

**PENGEMBANGAN APLIKASI WEB REKAM MEDIS STUDI  
KASUS RSIA. PROF. DR. H. M. FARID MAKASSAR**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:  
Nur Afdaliyah Anwar  
NIM: 145150201111065



PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2018

## PENGESAHAN

PENGEMBANGAN APLIKASI WEB REKAM MEDIS  
STUDI KASUS RSIA. PROF. DR. H. M. FARID MAKASSAR

### SKRIPSI


Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer


Disusun Oleh :  
Nur Afdaliyah Anwar  
NIM: 145150201111065

Skripsi ini telah diuji dan dinyatakan lulus pada  
3 Agustus 2018  
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I


Dosen Pembimbing II

  
Bayu Priyambadha, S.Kom, M.Kom  
NIP: 19820909 200812 1 004

  
Achmad Arwan, S.Kom, M.Kom  
NIP: 19840815 200812 1 004

Mengetahui  
Ketua Jurusan Teknik Informatika



  
Tri Astoto Kurniawan, S.T, M.T, Ph.D  
NIP: 19710518 200312 1 001

## PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 3 Agustus 2018



Nur Afdaliyah Anwar

NIM: 145150201111065



## KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Allah SWT yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga laporan skripsi yang berjudul “Pengembangan Aplikasi Web Rekam Medis Studi Kasus RSIA. Prof. Dr. H. M. Farid Makassar” ini dapat terselesaikan.

Penulis menyadari bahwa skripsi ini tidak akan berhasil tanpa bantuan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih banyak kepada:

1. Bapak Anwar Mursalin S dan Ibu Nurhana Husen selaku orang tua penulis, Chairil Anwar dan Mirza Alghifari Anwar selaku adik penulis, dan seluruh keluarga besar atas segala wejangan, perhatian, kasih sayang, dan kesabaran hati dalam membesarkan dan mendidik penulis, serta senantiasa memberikan doa dan semangat demi terselesainya skripsi ini,
2. Bapak Bayu Priyambadha, S.Kom., M.Kom. dan Bapak Achmad Arwan, S.Kom., M.Kom. selaku dosen pembimbing skripsi yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini,
3. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D. selaku Ketua Jurusan Teknik Informatika,
4. Anton J. Firdaus, Richa Amalia, Falihi Gozi, Adhy M, Nadya Ramadana yang sudah banyak membantu dan memberikan semangat kepada penulis,
5. Alfina Putri Said, Devy Muthia, Muthmainna Iskandar, Widya Kemalasari, Anshari Syakir, Arfandy Wijaya, Fazkya Zalika Amala yang selalu memberikan semangat dan dukungan moral kepada penulis dari jauh,
6. Talitha Raissa, Rona Salsabila, Niftah Fatihah, Tafarrara Irsa S, Cusen Mosabeth, dan Rifky Aufa yang senantiasa menemani dan memberikan dukungan,
7. Teman-teman seperjuangan penulis Nirzha Maulidya, Deandy Gilang, Abdi Wahid yang senantiasa mendengarkan keluh kesah penulis dan memberikan dukungan serta semangat,
8. Nirmala Saraswati, Isradi Azhar, dan Arsana Yudistira yang sudah sangat membantu di waktu-waktu kritis,
9. Ditya Enandini, Fendra Gunawan, Clara Pusparani yang senantiasa menghibur dan membantu penulis,
10. BE’M’ FILKOM, FARDIN, Dara Daeng Brawijaya, Nol Derajat yang juga memberikan banyak bantuan, dukungan, dan ilmu yang belum pernah penulis dapatkan sebelumnya,
11. Semua pihak yang tidak dapat disebutkan satu persatu, terima kasih banyak atas bantuan, semangat serta doa yang senantiasa diberikan kepada penulis.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih jauh dari kata sempurna, sehingga saran dan kritik yang membangun sangat penulis harapkan.

Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang membaca dan menggunakannya.

Malang, 22 Juli 2018

Penulis

nurafdaliyaha@gmail.com



## ABSTRAK

**Nur Afdaliyah A, Pembangunan Aplikasi Web Rekam Medis Studi Kasus Rsia.  
Prof. Dr. H. M. Farid Makassar**

**Pembimbing: Bayu Priyambadha, S.Kom., M.Kom. dan Achmad Arwan, S.Kom.**

Rumah sakit merupakan tempat perawatan, pelayanan, dan pemulihan penyakit serta tempat pengasuhan, bisa juga sebagai sarana pendidikan dan penelitian. Rumah Sakit Ibu dan Anak (RSIA) bila ditinjau dari klasifikasi rumah sakit, tergolong dalam rumah sakit Tipe C, dimana rumah sakit Tipe C ini hanya menyelenggarakan pelayanan kedokteran subspesialis terbatas, dalam hal ini pelayanan kesehatan anak dan pelayanan kebidanan dan kandungan. Pada RSIA. Prof. dr. H. M. Farid Makassar, dalam hal ini termasuk pendaftaran pasien dan pencatatan data rekam medis masih menggunakan media *worksheet*. Selain itu, proses pembuatan laporan juga masih dilakukan secara manual sehingga membuat pengolahan data pasien menjadi kurang maksimal. Untuk menangani permasalahan tersebut, dibangun sebuah aplikasi rekam medis berbasis web dengan fitur tambah dan cari data rekam medis untuk memudahkan pengolahan data pada RSIA. Prof. dr. H. M. Farid serta fitur untuk membuat laporan secara periodik. Penelitian ini menggunakan metode *waterfall* dan juga *prototype* dalam membangun perangkat lunak aplikasi rekam medis. Berdasarkan pengujian yang dilakukan yaitu pengujian unit yang menghasilkan nilai valid untuk semua jalur uji dan pengujian validasi yang menghasilkan nilai 100% valid pada semua kasus uji. Sedangkan pada pengujian *compatibility* yang diuji menggunakan *tool* SortSite menunjukkan bahwa sistem dapat berjalan dengan baik pada 9 dari 9 versi *browser* yang diujikan.

**Kata kunci:** aplikasi rekam medis, web, rekayasa perangkat lunak, model *waterfall*



## ABSTRACT

**Nur Afdaliyah A, Pembangunan Aplikasi Web Rekam Medis Studi Kasus Rsia.  
Prof. Dr. H. M. Farid Makassar**

**Advisor: Bayu Priyambadha, S.Kom., M.Kom. dan Achmad Arwan, S.Kom.**

*The hospital is a place of care, service, and recovery of illnesses as well as places of care, can also be a means of education and research. Maternal and Child Hospital (RSIA) when viewed from the hospital classification, is classified as a Type C hospital, where Type C hospitals only provide limited subspecialty medical services, in this case child health services and obstetrics and gynecological services. At RSIA. Prof. dr. H. M. Farid Makassar, in this case including patient registration and recording medical record data still using media worksheets. In addition, the report-making process is still carried out manually so as to make the processing of patient data less than optimal. To deal with these problems, a web-based medical record application with added features is built and search for medical record data to facilitate data processing at RSIA. Prof. dr. H. M. Farid and features to make periodic reports. This study uses the waterfall method and also a prototype in building medical record application software. Based on the tests carried out namely unit testing that produces valid values for all test lines and validation tests that produce 100% valid values in all test cases. While the compatibility testing tested using the SortSite tool shows that the system can run well in 9 of the 9 browser versions tested.*

**Keywords:** medical record application, web, software engineering, waterfall model

## DAFTAR ISI

PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT .....	vii
DAFTAR ISI .....	viii
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xix
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan .....	3
1.4 Manfaat.....	3
1.5 Batasan masalah .....	4
1.6 Sistematika pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN .....	6
2.1 Kajian Pustaka .....	6
2.2 Rekayasa Perangkat Lunak .....	6
2.3 Sistem Informasi Kesehatan .....	7
2.4 Rekam Medis .....	7
2.4.1 Pengisian Formulir Rekam Medis.....	7
2.5 Rekam Medis Elektronik .....	8
2.6 RSIA. Prof. Dr. H. M. Farid Makassar .....	8
2.6.1 Visi RSIA. Prof. Dr. H. M. Farid Makassar .....	9
2.6.2 Misi RSIA. Prof. Dr. H. M. Farid Makassar .....	9
2.6.3 Lokasi RSIA. Prof. Dr. H. M. Farid Makassar .....	9
2.6.4 Struktur Organisasi RSIA. Prof. Dr. H. M. Farid Makassar.....	9
2.7 Web.....	10
2.8 <i>Software Development Life Cycle (SDLC)</i> .....	10
2.8.1 Model <i>Waterfall</i> .....	10



2.8.2 Model <i>Prototype</i> .....	12
2.9 Database Management System (DBMS) .....	12
2.10 PHP .....	13
2.11 XAMPP .....	13
2.12 CodeIgniter .....	14
2.13 HTML .....	14
2.14 Pendekatan Berorientasi Objek .....	14
2.15 Unified Model Language .....	14
2.15.1 Class Diagram .....	15
2.15.2 Use Case Diagram .....	16
2.15.3 Sequence Diagram .....	17
2.16 Pengujian Perangkat Lunak .....	18
2.16.1 <i>White-Box Testing</i> .....	18
2.16.2 <i>Black-Box Testing</i> .....	19
BAB 3 METODOLOGI .....	20
3.1 Identifikasi Masalah .....	20
3.2 Studi Pustaka .....	21
3.3 Analisis Kebutuhan .....	21
3.3.1 Kebutuhan Fungsional .....	21
3.3.2 Kebutuhan Non Fungsional .....	22
3.4 Perancangan Sistem .....	22
3.5 Implementasi Sistem .....	22
3.6 Pengujian Sistem .....	22
3.7 Kesimpulan & Saran .....	22
BAB 4 ANALISIS KEBUTUHAN SISTEM .....	23
4.1 Analisis Permasalahan .....	23
4.1.1 Proses Bisnis pada RSIA. Prof. Dr. H. M. Farid Makassar .....	23
4.1.2 Dokumen Rekam Medis RSIA. Prof. Dr. H. M. Farid Makassar ...	24
4.2 Gambaran Umum Aplikasi .....	24
4.3 Identifikasi Aktor .....	24
4.4 Kebutuhan Fungsional .....	25
4.5 Kebutuhan Non Fungsional .....	37

4.6 Analisis Data.....	38
4.7 Pemodelan Kebutuhan Perangkat Lunak .....	39
4.7.1 <i>Use Case Diagram</i> .....	39
4.7.2 <i>Use Case Scenario</i> .....	40
BAB 5 PERANCANGAN DAN IMPLEMENTASI sistem .....	66
5.1 Perancangan Sistem.....	66
5.1.1 Perancangan Arsitektur.....	66
5.1.2 Perancangan Data .....	70
5.1.3 Perancangan Komponen .....	70
5.1.4 Perancangan Antarmuka.....	72
5.2 Implementasi Sistem .....	78
5.2.1 Spesifikasi Sistem .....	79
5.2.2 Implementasi Kode Program .....	79
5.2.3 Implementasi Data.....	82
5.2.4 Implementasi Antarmuka .....	83
BAB 6 PENGUJIAN SISTEM .....	91
6.1 Pengujian Unit.....	91
6.1.1 Pengujian Unit <i>Method</i> simpan_pendaftaran .....	91
6.1.2 Pengujian Unit <i>Method</i> simpan_anc.....	93
6.1.3 Pengujian Unit <i>Method</i> cetak_persalinan.....	95
6.2 Pengujian Validasi .....	97
6.2.1 Pengujian Validasi Login.....	97
6.2.2 Pengujian Validasi Logout .....	98
6.2.3 Pengujian Validasi Tambah Data Pendaftaran Pasien .....	98
6.2.4 Pengujian Validasi Lihat Daftar Data Pasien .....	99
6.2.5 Pengujian Validasi Edit Data Pendaftaran Pasien .....	99
6.2.6 Pengujian Validasi Cari Data Pasien .....	100
6.2.7 Pengujian Validasi Tambah Data Ringkasan Masuk dan Keluar	101
6.2.8 Pengujian Validasi Lihat Data Ringkasan Masuk dan Keluar.....	102
6.2.9 Pengujian Validasi Lihat Detail Data Ringkasan Masuk dan Keluar	102
6.2.10 Pengujian Validasi Edit Data Ringkasan Masuk dan Keluar ....	103

6.2.11 Pengujian Validasi Cari Data Ringkasan Masuk dan Keluar ....	103
6.2.12 Pengujian Validasi Tambah Data Resume Medis .....	104
6.2.13 Pengujian Validasi Lihat Data Resume Medis .....	105
6.2.14 Pengujian Validasi Lihat Detail Data Resume Medis.....	105
6.2.15 Pengujian Validasi Edit Data Resume Medis.....	105
6.2.16 Pengujian Validasi Cari Data Resume Medis.....	106
6.2.17 Pengujian Validasi Tambah Data Rencana Pulang .....	106
6.2.18 Pengujian Validasi Lihat Data Rencana Pulang .....	107
6.2.19 Pengujian Validasi Lihat Detail Data Rencana Pulang.....	108
6.2.20 Pengujian Validasi Edit Data Rencana Pulang.....	108
6.2.21 Pengujian Validasi Cari Data Rencana Pulang.....	109
6.2.22 Pengujian Validasi Tambah Data Pengkajian Asuhan Kebidanan .....	109
6.2.23 Pengujian Validasi Lihat Data Pengkajian Asuhan Kebidanan	110
6.2.24 Pengujian Validasi Lihat Detail Data Pengkajian Asuhan Kebidanan .....	110
6.2.25 Pengujian Validasi Edit Data Pengkajian Asuhan Kebidanan..	111
6.2.26 Pengujian Validasi Cari Data Pengkajian Asuhan Kebidanan..	111
6.2.27 Pengujian Validasi Tambah Data Pengkajian Asuhan Keperawatan .....	112
6.2.28 Pengujian Validasi Lihat Data Pengkajian Asuhan Keperawatan .....	113
6.2.29 Pengujian Validasi Lihat Detail Data Pengkajian Asuhan Keperawatan .....	113
6.2.30 Pengujian Validasi Edit Data Pengkajian Asuhan Keperawatan .....	114
6.2.31 Pengujian Validasi Cari Data Pengkajian Asuhan Keperawatan .....	114
6.2.32 Pengujian Validasi Tambah Data MR3.1 .....	115
6.2.33 Pengujian Validasi Lihat Data MR3.1 .....	116
6.2.34 Pengujian Validasi Lihat Detail Data MR3.1 .....	116
6.2.35 Pengujian Validasi Edit Data MR3.1 .....	116
6.2.36 Pengujian Validasi Cari Data MR3.1 .....	117
6.2.37 Pengujian Validasi Tambah Data MR3.2 .....	117

6.2.38 Pengujian Validasi Lihat Data MR3.2 .....	118
6.2.39 Pengujian Validasi Lihat Detail Data MR3.2 .....	118
6.2.40 Pengujian Validasi Edit Data MR3.2 .....	119
6.2.41 Pengujian Validasi Cari Data MR3.2 .....	119
6.2.42 Pengujian Validasi Tambah Data Rawat Jalan .....	119
6.2.43 Pengujian Validasi Lihat Data Rawat Jalan .....	120
6.2.44 Pengujian Validasi Lihat Detail Data Rawat Jalan .....	121
6.2.45 Pengujian Validasi Edit Data Rawat Jalan .....	121
6.2.46 Pengujian Validasi Cari Data Rawat Jalan .....	122
6.2.47 Pengujian Validasi Tambah Data Sebab Kematian .....	122
6.2.48 Pengujian Validasi Lihat Data Sebab Kematian.....	123
6.2.49 Pengujian Validasi Cetak Laporan Persalinan .....	123
6.2.50 Pengujian Validasi Cetak Laporan Kematian.....	124
6.2.51 Pengujian Validasi Tambah Data Visite.....	125
6.2.52 Pengujian Validasi Lihat Data Visite .....	126
6.2.53 Pengujian Validasi Lihat Detail Data Visite .....	126
6.2.54 Pengujian Validasi Edit Data Visite.....	126
6.2.55 Pengujian Validasi Cari Data Visite.....	127
6.2.56 Pengujian Validasi Tambah Data Dokter.....	127
6.2.57 Pengujian Validasi Lihat Daftar Data Dokter .....	128
6.2.58 Pengujian Validasi Edit Data Dokter .....	128
6.2.59 Pengujian Validasi Cari Data Dokter .....	128
6.3 Pengujian <i>Compatibility</i> .....	129
BAB 7 PENUTUP .....	130
7.1 Kesimpulan.....	130
7.2 Saran .....	130
DAFTAR PUSTAKA.....	131
LAMPIRAN A DATA DEFINITION LANGUAGE.....	133
LAMPIRAN B DATA FORMULIR REKAM MEDIS .....	142
LAMPIRAN C HASIL WAWANCARA DENGAN PIHAK RSIA .....	153

## DAFTAR TABEL

Tabel 2.1 Simbol-simbol pada <i>Class Diagram</i> .....	16
Tabel 2.2 Simbol-simbol pada <i>Use Case Diagram</i> .....	17
Tabel 2.3 Simbol-simbol pada <i>Sequence Diagram</i> .....	18
Tabel 4.1 Identifikasi Aktor .....	25
Tabel 4.2 Spesifikasi Kebutuhan Fungsional dan Pemetaan Use Case .....	25
Tabel 4.3 Kebutuhan Non Fungsional .....	37
Tabel 4.4 Use Case Scenario Login .....	41
Tabel 4.5 Use Case Scenario Logout .....	41
Tabel 4.6 Use Case Scenario Tambah Data Pendaftaran Pasien .....	41
Tabel 4.7 Use Case Scenario Lihat Daftar Data Pasien .....	42
Tabel 4.8 Use Case Scenario Edit Data Pendaftaran Pasien .....	42
Tabel 4.9 Use Case Scenario Cari Data Pasien .....	43
Tabel 4.10 Use Case Scenario Tambah Data Ringkasan Masuk dan Keluar .....	43
Tabel 4.11 Use Case Scenario Lihat Data Ringkasan Masuk dan Keluar .....	44
Tabel 4.12 Use Case Scenario Lihat Detail Data Ringkasan Masuk dan Keluar ....	44
Tabel 4.13 Use Case Scenario Edit Data Ringkasan Masuk dan Keluar Pasien .....	44
Tabel 4.14 Use Case Scenario Cari Data Ringkasan Masuk dan Keluar Pasien .....	45
Tabel 4.15 Use Case Scenario Tambah Data Resume Medis .....	45
Tabel 4.16 Use Case Scenario Lihat Data Resume Medis .....	46
Tabel 4.17 Use Case Scenario Lihat Detail Data Resume Medis .....	46
Tabel 4.18 Use Case Scenario Edit Data Resume Medis Pasien .....	46
Tabel 4.19 Use Case Scenario Cari Data Resume Medis Pasien .....	47
Tabel 4.20 Use Case Scenario Tambah Data Rencana Pulang .....	47
Tabel 4.21 Use Case Scenario Lihat Data Rencana Pulang .....	48
Tabel 4.22 Use Case Scenario Lihat Detail Data Rencana Pulang .....	48
Tabel 4.23 Use Case Scenario Edit Data Rencana Pulang Pasien .....	48
Tabel 4.24 Use Case Scenario Cari Data Rencana Pulang Pasien .....	49
Tabel 4.25 Use Case Scenario Tambah Data Pengkajian Asuhan Kebidanan .....	49
Tabel 4.26 Use Case Scenario Lihat Data Pengkajian Asuhan Kebidanan .....	50
Tabel 4.27 Use Case Scenario Lihat Detail Data Pengkajian Asuhan Kebidanan ..	50

Tabel 4.28 Use Case Scenario Edit Data Pengkajian Asuhan Kebidanan Pasien...	51
Tabel 4.29 Use Case Scenario Cari Data Pengkajian Asuhan Kebidanan Pasien ..	51
Tabel 4.30 Use Case Scenario Tambah Data Pengkajian Asuhan Keperawatan...	52
Tabel 4.31 Use Case Scenario Lihat Data Pengkajian Asuhan Keperawatan .....	52
Tabel 4.32 Use Case Scenario Lihat Detail Data Pengkajian Asuhan Keperawatan .....	53
Tabel 4.33 Use Case Scenario Edit Data Pengkajian Asuhan Keperawatan Pasien .....	53
Tabel 4.34 Use Case Scenario Cari Data Pengkajian Asuhan Keperawatan Pasien .....	54
Tabel 4.35 Use Case Scenario Tambah Data MR3.1 .....	54
Tabel 4.36 Use Case Scenario Lihat Data MR3.1.....	55
Tabel 4.37 Use Case Scenario Lihat Detail Data MR3.1 .....	55
Tabel 4.38 Use Case Scenario Edit Data MR3.1 Pasien.....	55
Tabel 4.39 Use Case Scenario Cari Data MR3.1 Pasien .....	56
Tabel 4.40 Use Case Scenario Tambah Data MR3.2 .....	56
Tabel 4.41 Use Case Scenario Lihat Data MR3.2.....	56
Tabel 4.42 Use Case Scenario Lihat Detail Data MR3.2 .....	57
Tabel 4.43 Use Case Scenario Edit Data MR3.2 Pasien.....	57
Tabel 4.44 Use Case Scenario Cari Data MR3.2 Pasien .....	57
Tabel 4.45 Use Case Scenario Tambah Data Rawat Jalan.....	58
Tabel 4.46 Use Case Scenario Lihat Data Rawat Jalan .....	58
Tabel 4.47 Use Case Scenario Lihat Detail Data Rawat Jalan .....	58
Tabel 4.48 Use Case Scenario Edit Data Rawat Jalan Pasien .....	59
Tabel 4.49 Use Case Scenario Cari Data Rawat Jalan Pasien .....	59
Tabel 4.50 Use Case Scenario Tambah Data Sebab Kematian.....	60
Tabel 4.51 Use Case Scenario Lihat Data Sebab Kematian .....	60
Tabel 4.52 Use Case Scenario Cetak Laporan Persalinan .....	60
Tabel 4.53 Use Case Scenario Cetak Laporan Kematian Bayi .....	61
Tabel 4.54 Use Case Scenario Tambah Data Visite .....	61
Tabel 4.55 Use Case Scenario Lihat Data Visite .....	62
Tabel 4.56 Use Case Scenario Lihat Detail Data Visite.....	62
Tabel 4.57 Use Case Scenario Edit Data Visite .....	62



Tabel 4.58 Use Case Scenario Cari Data Visite .....	63
Tabel 4.59 Use Case Scenario Tambah Data Dokter .....	63
Tabel 4.60 Use Case Scenario Lihat Daftar Data Dokter .....	64
Tabel 4.61 Use Case Scenario Edit Data Dokter .....	64
Tabel 4.62 Use Case Scenario Cari Data Dokter .....	64
Tabel 5.1 Algoritma <i>Method</i> simpan_pendaftaran .....	71
Tabel 5.2 Algoritma <i>Method</i> simpan_anc .....	71
Tabel 5.3 Algoritma <i>Method</i> cetak_persalinan .....	72
Tabel 5.4 Keterangan Antarmuka Halaman Login .....	73
Tabel 5.5 Keterangan Antarmuka Halaman Daftar Data Pasien .....	74
Tabel 5.6 Keterangan Antarmuka Halaman Pendaftaran Pasien .....	75
Tabel 5.7 Keterangan Antarmuka Halaman Data Rekam Medis .....	76
Tabel 5.8 Keterangan Antarmuka Halaman Tambah Data Rekam Medis .....	77
Tabel 5.9 Keterangan Antarmuka Halaman Laporan .....	78
Tabel 5.10 Spesifikasi Perangkat Keras .....	79
Tabel 5.11 Spesifikasi Perangkat Lunak .....	79
Tabel 5.12 Sistem Operasi .....	79
Tabel 5.13 Implementasi Kode Program <i>Method</i> simpan_pendaftaran .....	79
Tabel 5.14 Implementasi Kode Program <i>Method</i> simpan_anc .....	80
Tabel 5.15 Implementasi Kode Program <i>Method</i> cetak_persalinan .....	82
Tabel 5.16 <i>Data Definition Language</i> Aplikasi Rekam Medis RSIAPF .....	82
Tabel 6.1 <i>Pseudocode Method</i> simpan_pendaftaran .....	91
Tabel 6.2 Hasil Pengujian Unit <i>Method</i> simpan_pendaftaran .....	92
Tabel 6.3 <i>Pseudocode Method</i> simpan_anc .....	93
Tabel 6.4 Hasil Pengujian Unit <i>Method</i> simpan_anc .....	95
Tabel 6.5 <i>Pseudocode Method</i> cetak_persalinan .....	95
Tabel 6.6 Hasil Pengujian Unit <i>Method</i> cetak_persalinan .....	96
Tabel 6.7 Kasus Uji <i>Login</i> .....	97
Tabel 6.8 Kasus Uji <i>Login</i> Alternatif 1 .....	97
Tabel 6.9 Kasus Uji <i>Login</i> Alternatif 2 .....	97
Tabel 6.10 Kasus Uji <i>Logout</i> .....	98
Tabel 6.11 Kasus Uji Tambah Data Pendaftaran Pasien .....	98

Tabel 6.12 Kasus Uji Tambah Data Pendaftaran Pasien Alternatif 1 .....	98
Tabel 6.13 Kasus Uji Lihat Daftar Data Pasien .....	99
Tabel 6.14 Kasus Uji Edit Data Pendaftaran Pasien .....	99
Tabel 6.15 Kasus Uji Edit Data Pendaftaran Pasien Alternatif 1 .....	100
Tabel 6.16 Kasus Uji Cari Data Pasien .....	100
Tabel 6.17 Kasus Uji Cari Data Pasien Alternatif 1 .....	101
Tabel 6.18 Kasus Uji Tambah Data Ringkasan Masuk dan Keluar .....	101
Tabel 6.19 Kasus Uji Tambah Data Ringkasan Masuk dan Keluar Alternatif 1 ...	102
Tabel 6.20 Kasus Uji Lihat Data Ringkasan Masuk dan Keluar .....	102
Tabel 6.21 Kasus Uji Lihat Detail Data Ringkasan Masuk dan Keluar .....	102
Tabel 6.22 Kasus Uji Edit Data Ringkasan Masuk dan Keluar .....	103
Tabel 6.23 Kasus Uji Cari Data Ringkasan Masuk dan Keluar .....	103
Tabel 6.24 Kasus Uji Tambah Data Resume Medis .....	104
Tabel 6.25 Kasus Uji Tambah Data Resume Medis Alternatif 1 .....	104
Tabel 6.26 Kasus Uji Lihat Data Resume Medis .....	105
Tabel 6.27 Kasus Uji Lihat Detail Data Resume Medis .....	105
Tabel 6.28 Kasus Uji Edit Data Resume Medis .....	105
Tabel 6.29 Kasus Uji Cari Data Resume Medis .....	106
Tabel 6.30 Kasus Uji Tambah Data Rencana Pulang .....	106
Tabel 6.31 Kasus Uji Tambah Data Resume Medis Alternatif 1 .....	107
Tabel 6.32 Kasus Uji Lihat Data Rencana Pulang .....	107
Tabel 6.33 Kasus Uji Lihat Detail Data Rencana Pulang .....	108
Tabel 6.34 Kasus Uji Edit Data Rencana Pulang .....	108
Tabel 6.35 Kasus Uji Cari Data Rencana Pulang .....	109
Tabel 6.36 Kasus Uji Tambah Data Pengkajian Asuhan Kebidanan .....	109
Tabel 6.37 Kasus Uji Tambah Data Pengkajian Asuhan Kebidanan Alternatif 1.	110
Tabel 6.38 Kasus Uji Lihat Data Pengkajian Asuhan Kebidanan .....	110
Tabel 6.39 Kasus Uji Lihat Detail Data Pengkajian Asuhan Kebidanan .....	110
Tabel 6.40 Kasus Uji Edit Data Pengkajian Asuhan Kebidanan .....	111
Tabel 6.41 Kasus Uji Cari Data Pengkajian Asuhan Kebidanan .....	111
Tabel 6.42 Kasus Uji Tambah Data Pengkajian Asuhan Keperawatan .....	112

Tabel 6.43 Kasus Uji Tambah Data Pengkajian Asuhan Keperawatan Alternatif 1 .....	112
Tabel 6.44 Kasus Uji Lihat Data Pengkajian Asuhan Keperawatan .....	113
Tabel 6.45 Kasus Uji Lihat Detail Data Pengkajian Asuhan Keperawatan .....	113
Tabel 6.46 Kasus Uji Edit Data Pengkajian Asuhan Keperawatan.....	114
Tabel 6.47 Kasus Uji Cari Data Pengkajian Asuhan Keperawatan .....	114
Tabel 6.48 Kasus Uji Tambah Data MR3.1 .....	115
Tabel 6.49 Kasus Uji Tambah Data MR3.1 Alternatif 1 .....	115
Tabel 6.50 Kasus Uji Lihat Data MR3.1 .....	116
Tabel 6.51 Kasus Uji Lihat Data Detail MR3.1 .....	116
Tabel 6.52 Kasus Uji Edit Data MR3.1 .....	116
Tabel 6.53 Kasus Uji Cari Data MR3.1 .....	117
Tabel 6.54 Kasus Uji Tambah Data MR3.2 .....	117
Tabel 6.55 Kasus Uji Tambah Data MR3.2 Alternatif 1 .....	118
Tabel 6.56 Kasus Uji Lihat Data MR3.2 .....	118
Tabel 6.57 Kasus Uji Lihat Detail Data MR3.2 .....	118
Tabel 6.58 Kasus Uji Edit Data MR3.2 .....	119
Tabel 6.59 Kasus Uji Cari Data MR3.2 .....	119
Tabel 6.60 Kasus Uji Tambah Data Rawat Jalan.....	119
Tabel 6.61 Kasus Uji Tambah Data Rawat Jalan Alternatif 1 .....	120
Tabel 6.62 Kasus Uji Lihat Data Rawat Jalan .....	121
Tabel 6.63 Kasus Uji Lihat Detail Data Rawat Jalan .....	121
Tabel 6.64 Kasus Uji Edit Data Rawat Jalan .....	121
Tabel 6.65 Kasus Uji Cari Data Rawat Jalan .....	122
Tabel 6.66 Kasus Uji Tambah Data Sebab Kematian.....	122
Tabel 6.67 Kasus Uji Tambah Data Sebab Kematian Alternatif 1 .....	123
Tabel 6.68 Kasus Uji Lihat Data Sebab Kematian.....	123
Tabel 6.69 Kasus Uji Cetak Laporan Persalinan .....	123
Tabel 6.70 Kasus Uji Cetak Laporan Persalinan Alternatif 1 .....	124
Tabel 6.71 Kasus Uji Cetak Laporan Kematian.....	124
Tabel 6.72 Kasus Uji Cetak Laporan Kematian Alternatif 1 .....	125
Tabel 6.73 Kasus Uji Tambah Data Visite .....	125

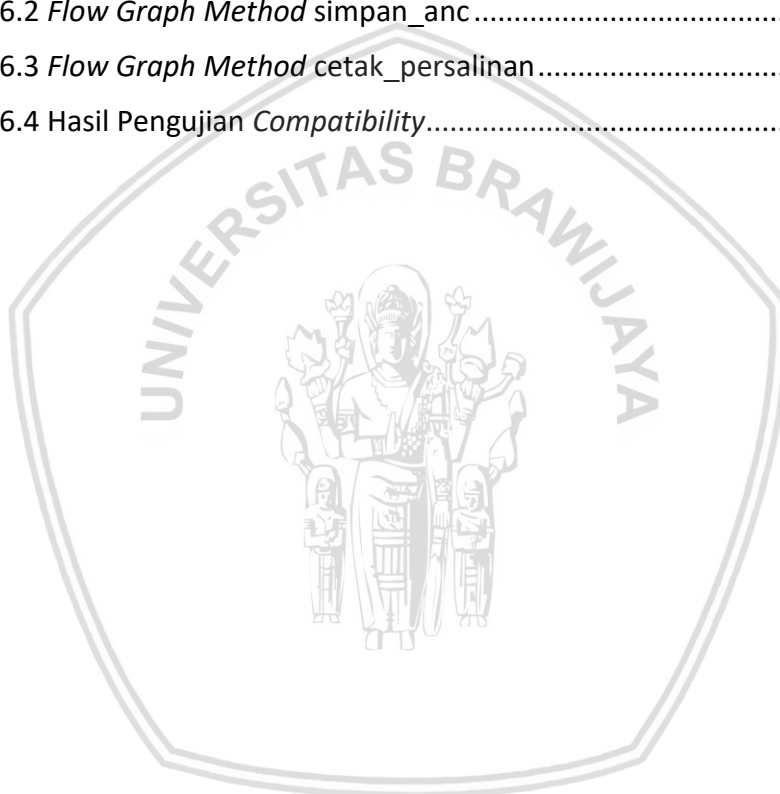
Tabel 6.74 Kasus Uji Lihat Data Visite .....	126
Tabel 6.75 Kasus Uji Lihat Detail Data Visite.....	126
Tabel 6.76 Kasus Uji Edit Data Visite.....	126
Tabel 6.77 Kasus Uji Cari Data Visite.....	127
Tabel 6.78 Kasus Uji Tambah Data Dokter.....	127
Tabel 6.79 Kasus Uji Lihat Daftar Data Dokter .....	128
Tabel 6.80 Kasus Uji Edit Data Dokter.....	128
Tabel 6.81 Kasus Uji Cari Data Dokter.....	128



## DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi RSIA. Prof. Dr. H. M. Farid .....	10
Gambar 2.2 Ilustrasi Model <i>Waterfall</i> .....	11
Gambar 2.3 Cara Kerja PHP .....	13
Gambar 2.4 Jenis Diagram UML .....	15
Gambar 3.1 Diagram Alur Metodologi Penelitian .....	20
Gambar 4.1 Diagram Proses Bisnis Rumah Sakit .....	23
Gambar 4.2 Entity Relationship Diagram Aplikasi Rekam Medis RSIAPF .....	38
Gambar 4.3 <i>Use Case Diagram</i> Aplikasi Rekam Medis RSIAPF .....	40
Gambar 5.1 <i>Sequence Diagram</i> Tambah Data Pendaftaran Pasien .....	66
Gambar 5.2 <i>Sequence Diagram</i> Tambah Data Rawat Jalan .....	67
Gambar 5.3 <i>Sequence Diagram</i> Cetak Laporan Persalinan .....	67
Gambar 5.4 Class Diagram Aplikasi Rekam Medis .....	69
Gambar 5.5 <i>Physical Data Model</i> Aplikasi Rekam Medis RSIAPF .....	70
Gambar 5.6 Antarmuka Halaman Login .....	73
Gambar 5.7 Antarmuka Halaman Daftar Data Pasien .....	74
Gambar 5.8 Antarmuka Halaman Pendaftaran Pasien .....	75
Gambar 5.9 Antarmuka Halaman Data Rekam Medis .....	76
Gambar 5.10 Antarmuka Halaman Tambah Data Rekam Medis .....	77
Gambar 5.11 Antarmuka Halaman Laporan .....	78
Gambar 5.12 Implementasi Antarmuka Halaman Login .....	83
Gambar 5.13 Implementasi Antarmuka Halaman Beranda .....	83
Gambar 5.14 Implementasi Antarmuka Halaman Daftar Data Pasien .....	84
Gambar 5.15 Implementasi Antarmuka Halaman Pendaftaran Pasien .....	84
Gambar 5.16 Implementasi Antarmuka Halaman Data Rekam Medis .....	85
Gambar 5.17 Potongan Antarmuka Halaman Tambah Data Ringkasan Masuk dan Keluar .....	85
Gambar 5.18 Potongan Antarmuka Halaman Tambah Data Resume Medis .....	86
Gambar 5.19 Potongan Antarmuka Halaman Tambah Data Rencana Pulang .....	86
Gambar 5.20 Potongan Antarmuka Halaman Tambah Data Pengkajian Asuhan Kebidanan .....	87

Gambar 5.21 Potongan Antarmuka Halaman Tambah Data Pengkajian Asuhan Keperawatan .....	87
Gambar 5.22 Potongan Antarmuka Halaman Tambah Data MR3.1.....	88
Gambar 5.23 Potongan Antarmuka Halaman Tambah Data MR3.2.....	88
Gambar 5.24 Potongan Antarmuka Halaman Tambah Data Rawat Jalan .....	89
Gambar 5.25 Potongan Antarmuka Halaman Tambah Data Sebab Kematian .....	89
Gambar 5.26 Implementasi Antarmuka Halaman Laporan .....	90
Gambar 6.1 <i>Flow Graph Method</i> simpan_pendaftaran.....	92
Gambar 6.2 <i>Flow Graph Method</i> simpan_anc .....	94
Gambar 6.3 <i>Flow Graph Method</i> cetak_persalinan .....	96
Gambar 6.4 Hasil Pengujian <i>Compatibility</i> .....	129





## BAB 1 PENDAHULUAN

### 1.1 Latar Belakang

Pada zaman ini, teknologi informasi berkembang dengan pesat, perkembangan teknologi sangat mempengaruhi aktivitas manusia. Dengan dukungan sarana teknologi yang canggih, maka berbagai jenis informasi lebih mudah untuk didapatkan. Salah satu teknologi yang sangat membantu manusia adalah komputer. Komputer merupakan suatu alat yang bisa digunakan untuk menyelesaikan sebuah masalah. Komputer memiliki dua komponen yaitu *software* dan *hardware* yang merupakan media paling cepat serta akurat dalam mengolah data. Perkembangan teknologi dapat direalisasikan kegunaannya apabila didukung oleh sumber daya manusia yang baik.

Salah satu bidang ilmu yang terkena dampak positif dari perkembangan teknologi adalah dunia kesehatan. Tanpa kita sadari dunia kesehatan merupakan salah satu faktor yang penting dalam aspek kehidupan. Seiring berjalannya waktu menuju dunia modern, dunia kesehatan sudah seharusnya memberi pelayanan yang cepat dan tepat. Pada situasi yang cukup meprihatinkan saat ini, dunia kesehatan seharusnya bisa menciptakan dan mewujudkan lingkungan kerja yang kondusif, nyaman, serta penuh motivasi sehingga dapat memberi pelayanan kesehatan yang memuaskan.

Menurut *World Health Organization* (WHO), rumah sakit adalah sebuah instansi perawatan kesehatan profesional yang merupakan bagian integral dari sebuah organisasi sosial dan kesehatan yang pelayanannya dilakukan oleh dokter, perawat, dan tenaga ahli kesehatan lainnya dengan fungsi menyediakan pelayanan paripurna (*comprehensive*), penyembuhan penyakit (*curative*), dan pencegahan penyakit (*preventive*). Rumah sakit merupakan salah satu sarana kesehatan dimana tempat untuk menyelenggarakan upaya kesehatan dengan memanfaatkan berbagai kesatuan anggota terlatih dan terdidik untuk menghadapi dan menangani masalah medis baik untuk pemulihan dan pemeliharaan kesehatan. Berdasarkan UU No 44 tahun 2009 tentang Rumah Sakit, menyatakan bahwa fungsi rumah sakit yaitu menyelenggarakan pelayanan pengobatan dan pemulihan kesehatan sesuai dengan standar pelayanan rumah sakit. Berdasarkan definisi rumah sakit yang telah dijelaskan diatas, dapat disimpulkan bahwa kegunaan rumah sakit tidak hanya sebagai tempat perawatan, pelayanan, dan pemulihan penyakit melainkan juga tempat pengasuhan, sarana pendidikan dan juga penelitian.

Pelayanan rumah sakit tersebut ditujukan kepada pasien, masyarakat luas, dan juga institusi atau lembaga. Pelayanan yang ditujukan kepada pasien meliputi pemeriksaan, diagnosa, tindakan (pengobatan atau pembedahan). Pelayanan rumah sakit terbagi atas pelayanan dasar yang meliputi rawat jalan yang merupakan pertolongan untuk penderita yang masih berada pada kategori cukup sehat untuk bias kembali ke rumah, rawat inap yang merupakan pertolongan untuk penderita yang memerlukan perawatan khusus secara berkelanjutan hingga penderita dikatakan pulih dan sembuh, dan unit gawat darurat yang merupakan

pemberian pertolongan untuk penderita yang berada dalam kategori darurat dan harus segera mendapat tindakan lanjut. Selanjutnya pelayanan spesialis yang meliputi pelayanan spesialis bedah, penyakit dalam, keuseran dan penyakit kandungan, kesehatan anak, dan sebagainya. Kemudian yang disebut pelayanan penunjang seperti radiologi, laboratorium, farmasi, dan kamar operasi. Adapun bentuk pelayanan tersebut dapat ditentukan berdasarkan tipe rumah sakit.

Rumah Sakit Ibu dan Anak (RSIA) jika ditinjau dari kalsifikasi rumah sakit, tergolong dalam rumah sakit Tipe C, dimana rumah sakit Tipe C ini hanya menyelenggarakan pelayanan kedokteran subspesialis terbatas, dalam hal ini pelayanan kesehatan anak dan pelayanan kebidanan dan kandungan. Layanan pada Rumah Sakit Ibu dan Anak (RSIA) terbagi atas pelayanan preventif, kuratif, dan rehabilitasi. Pelayanan preventif merupakan pelayanan yang bertujuan untuk mencegah pasien agar tidak terjangkit dari penyakit, dan dapat dilakukan dengan cara pemeriksaan rutin terhadap perkembangan bayi dan ibu hamil, konsultasi kesehatan, penyuluhan tentang gizi ibu dan anak, serta imunisasi dan KB. Pelayanan kuratif merupakan pelayanan yang bertujuan untuk menyembuhkan pasien melalui pengobatan dan perawatan berupa persalinan, pembedahan, dan pengobatan. Serta pelayanan rehabilitasi yang merupakan pelayanan yang bertujuan menyembuhkan kondisi fisik pasien setelah melampaui masa pengobatan berupa perawatan atau pemulihan kesehatan ibu dan perawatan bayi.

RSIA. Prof. Dr. H. M. Farid Makassar merupakan salah satu rumah sakit swasta yang sedang berkembang dan berusaha untuk memberikan pelayanan yang baik dan maksimal terhadap masyarakat, khususnya pelayanan terhadap ibu hamil, melahirkan, dan anak. Tetapi untuk poli anak itu sendiri masih belum terlaksana. Sistem pencatatan rekam medis pasien RSIA. Prof. Dr. H. M. Farid Makassar masih menggunakan media *worksheet* dalam pengolahan data seperti pendaftaran pasien, pencatatan rekam medis, dan pembuatan laporan. Dengan menggunakan sistem tersebut mengakibatkan pekerjaan menjadi kurang optimal.

Pelayanan yang baik adalah pelayanan yang menguntungkan bagi pasien, pihak rumah sakit, dan juga pihak yang terkait. Untuk meningkatkan mutu pelayanan dapat dengan menciptakan pelayanan yang cepat, tepat, serta akurat, baik pada pelayanan medis maupun non-medis. Adapun peran rekam medis dalam pelayanan rumah sakit yaitu sebagai administrasi dimana data dan informasi dapat digunakan oleh manajemen untuk melaksanakan fungsinya dalam mengelola berbagai sumber daya. Sebagai *research* dimana dari data-data yang ada pada rekam medis, dapat dilakukan penelusuran terhadap berbagai macam penyakit dan sejenisnya. Sebagai *education* yang dapat digunakan untuk pengembangan ilmu. Dan juga sebagai dokumentasi untuk menyimpan sejarah medis seseorang.

Berdasarkan permasalahan diatas, maka diambil judul “Pengembangan Aplikasi Web Rekam Medis Studi Kasus RSIA. Prof. Dr. H. M Farid Makassar” yang merupakan sebuah solusi berupa perangkat lunak komputer dan diharapkan dari pengolahan data tersebut dapat diperoleh output/keluaran berupa informasi yang akurat dan relevan, serta data-data yang ada dapat diolah secara efisien dan tepat

sehingga dapat menghasilkan informasi yang bisa dipertanggungjawabkan kebenarannya.

## 1.2 Rumusan masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka penulis dapat merumuskan masalah sebagai berikut:

1. Bagaimana analisis kebutuhan sistem yang akan digunakan pada RSIA. Prof. Dr. H. M Farid Makassar?
2. Bagaimana perancangan sistem yang akan digunakan pada RSIA. Prof. Dr. H. M Farid Makassar?
3. Bagaimana implementasi sistem yang akan digunakan pada RSIA. Prof. Dr. H. M Farid Makassar?
4. Bagaimana pengujian sistem yang akan digunakan pada RSIA. Prof. Dr. H. M Farid Makassar?

## 1.3 Tujuan

Penelitian ini bertujuan untuk:

1. Melakukan dan mengetahui analisis kebutuhan sistem yang akan digunakan pada RSIA. Prof. Dr. H. M Farid Makassar.
2. Melakukan dan mengetahui perancangan sistem yang akan digunakan pada RSIA. Prof. Dr. H. M Farid Makassar.
3. Melakukan dan mengetahui implementasi sistem yang akan digunakan pada RSIA. Prof. Dr. H. M Farid Makassar.
4. Melakukan dan mengetahui pengujian sistem yang akan digunakan pada RSIA. Prof. Dr. H. M Farid Makassar.

## 1.4 Manfaat

Manfaat dari penelitian ini adalah sebagai berikut:

1. Bagi RSIA. Prof. dr. H. M Farid Makassar

Sistem ini diharapkan dapat mendukung kegiatan proses pengolahan data pasien.

2. Bagi penulis

Dengan adanya pembangun sistem ini, diharapkan dapat menambah ilmu wawasan serta pengalaman penelitian, serta mampu mengimplementasikan ilmu yang telah didapat selama menjadi mahasiswa Fakultas Ilmu Komputer.

## 1.5 Batasan masalah

Agar pembahasan tidak mengarah ke penjelasan yang lebih luas, maka penelitian ditekankan pada hal-hal berikut:

1. Aplikasi ini terbatas pada pengolahan data rekam medis pasien yang berkunjung ke RSIA. Prof. Dr. H. M Farid Makassar.
2. Jenis laporan yang ada didalam aplikasi hanya laporan persalinan dan laporan kematian.
3. Data yang diperoleh merupakan hasil wawancara dengan bagian rekam medis rumah sakit dan pihak yang berwenang pada RSIA. Prof. Dr. H. M Farid Makassar.

## 1.6 Sistematika pembahasan

Sistematika penyusunan laporan ini dilampirkan untuk memberikan gambaran serta uraian dari penelitian ini secara garis besar yang meliputi beberapa bab, sebagai berikut:

### **BAB I Pendahuluan**

Pada bab ini, berisi tentang latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, metodologi penelitian, dan juga sistematika penulisan.

### **BAB II Landasan Kepustakaan**

Pada bab ini, berisi tentang kajian pustaka serta landasarn teori yang akan digunakan sebagai dasar untuk pembangunan aplikasi rekam medis berbasis web.

### **BAB III Metodologi Penelitian**

Pada bab ini, berisi tentang metode yang digunakan dalam penelitian ini dan terdiri dari studi literatur, perancangan sistem perangkat lunak, implementasi perangkat lunak, pengujian dan analisis, serta penulisan laporan.

### **BAB IV Analisa Kebutuhan Sistem**

Pada bab ini, berisi tentang uraian mengenai analisa kebutuhan sistem yang mencakup pemodelan sistem dengan UML (*Unified Model Language*).

### **BAB V Perancangan dan Implementasi Sistem**

Pada bab ini, berisi tentang perancangan sistem dan perancangan basis data serta implementasi sistem sesuai dengan analisis kebutuhan sistem yang telah dibuat.

### **BAB VI Pengujian Sistem**

Pada bab ini, berisi tentang hasil pengujian terhadap sistem yang telah direalisasikan.

## BAB VII Penutup

Pada bab ini, berisi kesimpulan yang diperoleh dari perancangan dan pengujian perangkat lunak yang dikembangkan dalam skripsi ini serta kritik dan saran untuk penelitian selanjutnya.



## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Kajian Pustaka

Referensi pertama, berdasarkan penelitian yang telah dilakukan oleh saudara Bagus An Arif dengan judul “Aplikasi Rekam Medis Pasien Berbasis Web pada Klinik Belinda Rajeg Kab. Tangerang”. Penelitian tersebut bertujuan untuk mengetahui sistem pengolahan rekam medis pada Klinik Belinda, agar memberikan informasi yang cepat, tepat, serta relevan, dan juga untuk membangun aplikasi rekam medis yang dapat mempermudah dalam mengelola data pasien secara maksimal. Hasil dari penelitian tersebut adalah suatu sistem yang memiliki keluaran/*output* berupa informasi yang dapat digunakan untuk pengambilan keputusan atau hal lain yang dapat menjadi *input* bagi subsistem lain (Arief, 2016).

Referensi kedua, berdasarkan penelitian yang telah dilakukan oleh saudari Alqarana Aljufri dengan judul “Aplikasi Rekam Medis Studi Kasus Klinik Universitas Widyatama”. Penelitian tersebut bertujuan untuk membangun sebuah aplikasi yang dapat mempermudah pendaftaran pasien dan mempermudah dalam membuat pelaporan rekam medis. Hasil dari penelitian tersebut merupakan sebuah aplikasi rekam medis yang dapat digunakan pada Klinik Universitas Widyatama (Aljufri, 2013).

Aplikasi rekam medis pada penelitian ini menggunakan OOP (*Object Oriented Programming*) dengan model *waterfall* dan *prototype*. Model *waterfall* digunakan karena aplikasi yang akan dibangun sudah jelas kebutuhannya diawal sehingga dapat meminimalisir kesalahan. Model *prototype* digunakan karena *user* hanya mendefinisikan secara umum terhadap apa yang harus dilakukan sebuah sistem tanpa menyebutkan keluaran apa yang dibutuhkan, pemrosesan serta data apa saja yang dibutuhkan.

### 2.2 Rekayasa Perangkat Lunak

Perangkat lunak merupakan sebuah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, serta tata cara penggunaan (*user manual*). Sebuah program komputer yang tidak terasosiasi dengan dokumentasinya belum bisa disebut sebagai *software* (Shalahuddin & S, 2013).

Rekayasa Perangkat Lunak (*Software Engineering*) merupakan sebuah pembangunan dengan menggunakan konsep rekayasa yang bertujuan menghasilkan perangkat lunak yang bernilai ekonomi dan dipercaya serta dapat bekerja dengan efisien menggunakan mesin. Rekayasa perangkat lunak difokuskan pada bagaimana membangun perangkat lunak yang dapat terus menerus dipelihara setelah perangkat lunak selesai dibangun seiring dengan berjalannya waktu dan berkembangnya teknologi serta lingkungan (*maintainability*), dapat diandalkan sesuai dengan proses bisnis yang dijalankan serta perubahan yang terjadi (*dependability* dan *robust*), efisien terhadap sumber daya dan juga



penggunaan, serta kemampuan untuk dapat digunakan sesuai kebutuhan (*usability*).

Dari kriteria yang telah disebutkan, dapat disimpulkan bahwa perangkat lunak yang baik yaitu perangkat lunak yang dapat memenuhi kebutuhan *costumer* atau *user* ataupun yang berorientasi pada pemakai perangkat lunak, bukan berorientasi pada pengembang perangkat lunak (Shalahuddin & S, 2013).

## 2.3 Sistem Informasi Kesehatan

Sistem informasi merupakan sekumpulan dari perangkat lunak, perangkat keras, *user*, prosedur serta aturan yang diorganisasikan secara utuh untuk mengolah data menjadi sebuah informasi yang berguna untuk memecahkan masalah serta pengambilan keputusan. Sistem informasi juga merupakan perpaduan antara sistem manusia atau mesin untuk menyajikan informasi guna mendukung fungsi operasi, manajemen, serta pengambilan keputusan (McLeod, 2008).

Sistem informasi kesehatan merupakan sebuah sistem informasi yang terdiri dari manusia, proses, dan data, serta proses dan kombinasi antara perangkat lunak, perangkat keras dan teknologi informasi. Penggunaan sistem informasi terlibat dalam tiga tahap yaitu pemasukan data, pemrosesan data, dan pengeluaran informasi (Hatta, 2012).

## 2.4 Rekam Medis

Menurut Peraturan Menteri Kesehatan Nomor 55 tahun 2013, rekam medis merupakan sebuah berkas yang berisikan dokumen dan catatan mengenai identitas pasien pemeriksaan, diagnosa, tindakan, dan pelayanan lainnya yang diberikan kepada pasien pada fasilitas pelayanan kesehatan. Rekam medis bersifat rahasia dan dimiliki oleh pasien. Tujuan rekam medis yaitu sebagai penunjang tertib administrasi dalam upaya peningkatan pelayanan kesehatan di rumah sakit.

Peraturan Menteri Kesehatan Nomor 269 tahun 2009, menyatakan bahwa rekam medis harus mampu menyediakan informasi tentang pelayanan dan pengobatan di rumah sakit. Pembuatan rekam medis harus lengkap dan jelas dimana rekam medis tersebut dapat dilakukan secara tertulis maupun secara elektronik. Rekam medis juga berisi tentang semua tindakan yang diberikan terhadap pasien, baik pasien rawat jalan, rawat inap, dan gawat darurat.

### 2.4.1 Pengisian Formulir Rekam Medis

Beberapa hal yang perlu diperhatikan pada pengisian formulir rekam medis adalah sebagai berikut:

#### 2.4.1.1 Prosedur Pengisian

Setiap catatan dalam rekam medis harus dilengkapi dengan nama, waktu, petugas yang memberikan pelayanan ataupun tindakan. Dalam Pedoman Pengelolaan Rekam Medis Rumah Sakit di Indonesia (1997), dijelaskan bahwa

rekam medis harus dengan segera dilengkapi sesaat setelah pasien menerima pelayanan. Semua pencatatan harus ditandatangani oleh dokter atau tenaga kesehatan yang sesuai kewenangannya dan diberi tanggal. Penghapusan dengan cara apapun tidak diperbolehkan.

#### **2.4.1.2 Penanggungjawab Pengisian Rekam Medis**

Dalam buku Pedoman Pengelolaan Rekam Medis Rumah Sakit di Indonesia (1997), yang berhak membuat atau mengisi rekam medis adalah tenaga kesehatan yang memberikan pelayanan langsung kepada pasien atas izin atau perintah dari dokter dan dokter gigi yang menjalankan praktik kedokteran. Tenaga *paramedic* perawatan dan tenaga *paramedic* non-perawatan yang langsung terlibat didalamnya antara lain adalah perawat, perawat gigi, user, tenaga laboratorium klinik, gizi, anastesi, penata rontgen, dan, rehabilitasi medik.

#### **2.4.1.3 Kelengkapan Pengisian Rekam Medis**

Sesuai dengan Keputusan Menteri Kesehatan Nomor 129 tahun 2009 tentang Standar Pelayanan Minimal Rumah Sakit, kelengkapan pengisian rekam medis adalah 24 jam setelah pelayanan selesai dilakukan. Hal ini dimaksudkan supaya terjadi sinkronisasi pelayanan dan keselamatan, yang bertujuan tergambarnya tanggung jawab dokter pada kelengkapan informasi dalam rekam medis. Rekam medis yang telah diisi lengkap dalam waktu kurang dari 24 jam setelah pelayanan rawat jalan selesai atau setelah pasien rawat inap ditetapkan untuk pulang dengan standar pengisian terisi penuh yang meliputi identitas pasien, anamnesis, rencana asuhan, pelaksanaan asuhan, tindak lanjut, dan resume.

### **2.5 Rekam Medis Elektronik**

Rekam Medis Elektronik (RME) merupakan sistem yang menangani penyimpanan data klinis, entry data yang terkomputerisasi, serta dokumentasi medis. Rekam Medis Elektronik (RME) berguna bagi tenaga medis sebagai penyelenggara layanan kesehatan untuk mendokumentasikan, memonitor, serta mengelola pelayanan kesehatan yang diberikan kepada pasien baik di rumah sakit dan unit layanan kesehatan lainnya, misalnya praktik pribadi dokter. Secara hukum, data dalam RME merupakan rekaman legal yang didapatkan dari pelayanan yang telah diberikan pada pasien dan rumah sakit juga memiliki hak untuk menyimpan data tersebut (Wimmie, 2009).

Pada Rekam Medis Elektronik, harus dipastikan bahwa data dan informasi tentang pasien yang bersifat rahasia tersimpan dengan aman dan hanya dapat diakses oleh pihak yang berkepentingan, dalam hal ini dokter yang menuliskan rekam medis pasien, dan tentu saja pasien pemilik data rekam medis tersebut.

### **2.6 RSIA. Prof. Dr. H. M. Farid Makassar**

RSIA. Prof. Dr. H. M. Farid berdiri diatas tanah seluas 1.613 m<sup>2</sup>, dengan luas bangunan 1.885 m<sup>2</sup> dan terdiri dari 2 lantai. Bangunan tersebut difungsikan untuk

IGD, Poliklinik, Apotek, Ruang Perawatan, Kamar Bersalin, Kamar Operasi, Kamar Bayi, dan Kantor Administrasi.

### **2.6.1 Visi RSIA. Prof. Dr. H. M. Farid Makassar**

Visi RSIA. Prof. Dr. H. M. Farid Makassar adalah sebagai berikut :

- Menjadi Rumah Sakit yang terpercaya dan dambaan masyarakat

### **2.6.2 Misi RSIA. Prof. Dr. H. M. Farid Makassar**

Misi RSIA. Prof. Dr. H. M. Farid Makassar adalah sebagai berikut :

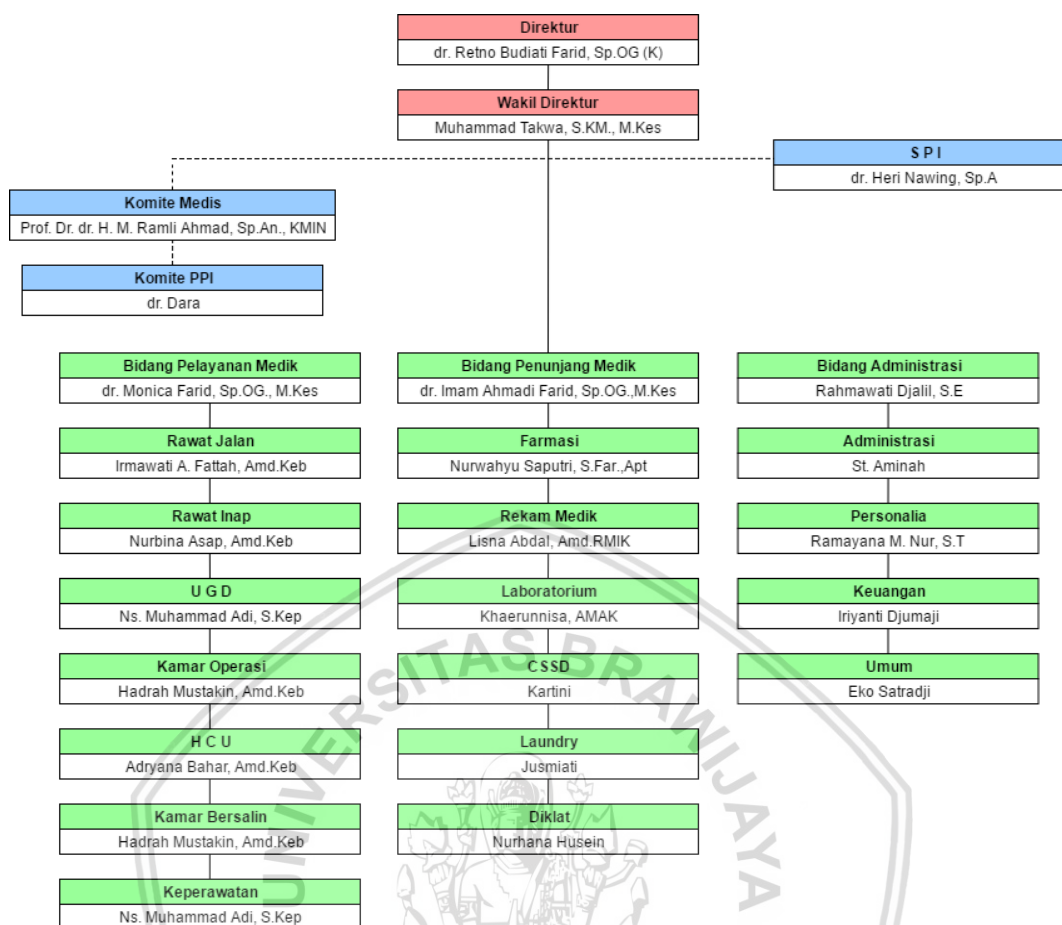
- Memberikan pelayanan yang profesional dan aman
- Meningkatkan kemampuan profesional SDM Rumah Sakit
- Mengembangkan dan meningkatkan teknologi sarana dan prasarana Rumah Sakit
- Melaksanakan pola pembiayaan yang efektif, efisien, dan terjangkau
- Upaya meningkatkan kesejahteraan karyawan

### **2.6.3 Lokasi RSIA. Prof. Dr. H. M. Farid Makassar**

Lokasi RSIA. Prof. Dr. H. M. Farid terletak di Kota Makassar, Sulawesi Selatan tepatnya di Jalan Dr. Wahidin Sudirohusodo No. 230, Kelurahan Melayu, Kecamatan Wajo.

### **2.6.4 Struktur Organisasi RSIA. Prof. Dr. H. M. Farid Makassar**

Susunan struktur organisasi RSIA. Prof. Dr. H. M. Farid dijabarkan pada Gambar 2.1 berikut yang dipimpin oleh seorang direktur dan wakil direktur. Selain itu, terdapat 3 komite yaitu komite medik, komite PPI, dan SPI, serta terdapat 3 bidang pelayanan yaitu pelayanan medik yang meliputi rawat jalan, rawat inap, UGD, kamar operasi, HCU, kamar bersalin, dan keperawatan. Bidang penunjang medik yang meliputi farmasi, rekam medik, laboratorium, cssd, *laundry*, dan diklat. Serta bidang administrasi, yang meliputi administrasi, personalia, keuangan, dan umum. Pada penelitian kali ini berfokus pada bidang pelayanan medic, serta bidang penunjang medik bagian rekam medik.



**Gambar 2.1 Struktur Organisasi RSIA. Prof. Dr. H. M. Farid**  
(Dika, 2014)

## 2.7 Web

Web atau yang dikenal dengan *World Wide Web* yang disingkat menjadi *WWWi* merupakan sebuah layanan yang dapat digunakan oleh pengguna komputer untuk terhubung ke internet. Web menyediakan berbagai macam informasi. *Website* dapat juga diartikan sebagai sekumpulan halaman yang berfungsi untuk menampilkan informasi berbentuk teks, animasi, suara, gambar ataupun gabungan dari semuanya, dapat bersifat statis maupun dinamis yang dapat membentuk sebuah rangkaian yang saling terkait satu sama lain (Simamarta, 2010).

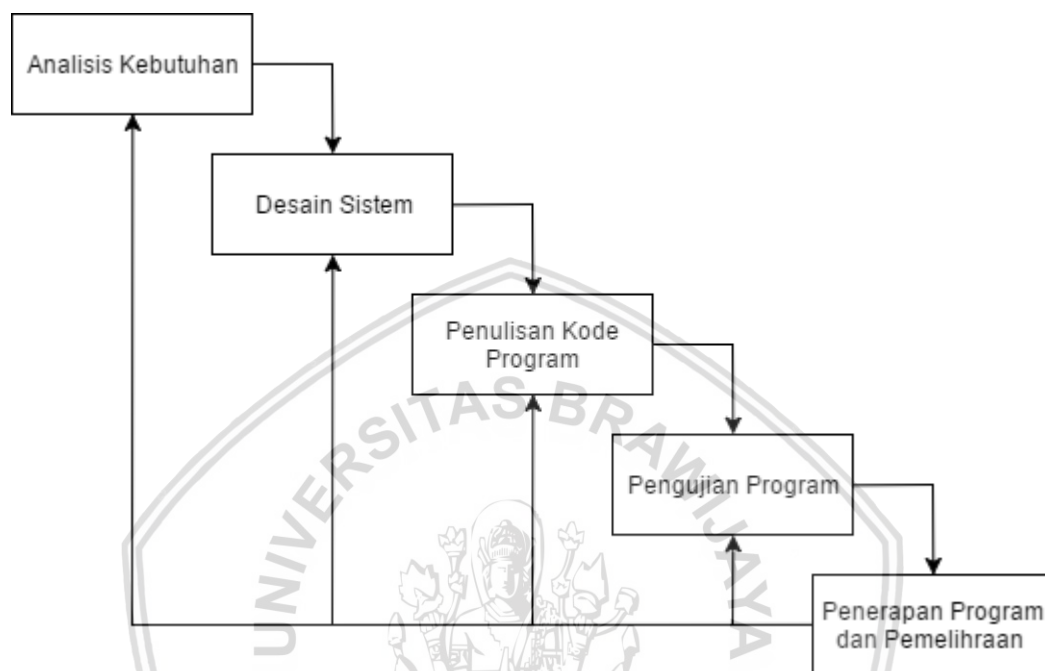
## 2.8 Software Development Life Cycle (SDLC)

SDLC merupakan tahap untuk mengembangkan atau mewujudkan suatu perangkat lunak dengan menggunakan model ataupun metodologi. Model SDLC dasar yang digunakan adalah model *waterfall* dan *prototype*

### 2.8.1 Model Waterfall

Model air terjun atau yang seringkali dikenal dengan sebutan model *waterfall* dan juga dikenal dengan model sekuensial linier (*sequential linear*) atau alur hidup

klasik (*classic life cycle*). Model *waterfall* ini menyediakan pendekatan alur hidup perangkat lunak secara berurut dimulai dengan proses analisis kebutuhan, desain sistem, penulisan kode program, pengujian program, dan penerapan program dan pemeliharaan. Dengan menggunakan model *waterfall*, pengembang dan *stakeholder* harus merencanakan semua proses kegiatan dari awal (Shalahuddin & S, 2013).



**Gambar 2.2 Ilustrasi Model *Waterfall***

(Shalahuddin & S, 2013)

Pada Gambar 2.2 diatas menunjukkan ilustrasi dengan model *waterfall* dengan penjelasan setiap tahapnya adalah sebagai berikut:

#### **2.8.1.1 Analisis Kebutuhan**

Tahap ini merupakan proses pengumpulan kebutuhan yang dilakukan secara rutin untuk mendefinisikan kebutuhan perangkat lunak supaya dapat dipahami oleh *user* (pengguna). Analisis persyaratan, definisi layanan sistem, kendala serta tujuan ditetapkan pada tahap ini melalui konsultasi juga dengan pengguna sistem (Shalahuddin & S, 2013).

#### **2.8.1.2 Desain Sistem**

Tahap ini merupakan proses *multistep* yang berfokus pada desain pembuatan program perangkat lunak yang meliputi struktur data, representasi antarmuka (*interface*), arsitektur perangkat lunak, serta prosedur penulisan kode program. Tahap ini menerjemahkan kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan pada tahap selanjutnya (Shalahuddin & S, 2013).



### 2.8.1.3 Penulisan Kode Program

Tahap ini merupakan proses mentranslasi desain yang sudah dibuat ke dalam bentuk kode. Hasil dari tahap ini berupa sebuah program pada komputer yang sesuai dengan hasil desain yang telah dibuat pada tahap sebelumnya (Shalahuddin & S, 2013).

### 2.8.1.4 Pengujian Program

Tahap ini merupakan tahap untuk menguji perangkat lunak dari segi *logic* maupun fungsional, dan juga memastikan bahwa semua unit dari perangkat lunak telah diuji dengan baik. Hal ini dilakukan agar kesalahan (*error*) dapat diminimalisir dan memastikan bahwa keluaran (*output*) sesuai dengan yang diinginkan (Shalahuddin & S, 2013).

### 2.8.1.5 Penerapan Program dan Pemeliharaan

Tahap ini merupakan tahap akhir pada model *waterfall*. Dari semua tahap yang telah dikerjakan, ada kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah diterbitkan dan dikirim ke *user*. Perubahan-perubahan bisa terjadi karena terdapat kesalahan yang tidak terdeteksi saat pengujian atau mungkin karena perangkat lunak harus beradaptasi dengan lingkungan baru. Pada tahap ini, proses pengembangan dapat dimulai lagi dari tahap analisis spesifikasi untuk perubahan perangkat lunak yang sudah dibangun sebelumnya, tetapi tidak untuk membuat sebuah perangkat lunak yang baru (Shalahuddin & S, 2013).

## 2.8.2 Model *Prototype*

Model *prototype* digunakan untuk menjelaskan ketidakpahaman pelanggan mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pelanggan kepada pengembang perangkat lunak. Model *prototype* dimulai dengan mengumpulkan kebutuhan user terhadap perangkat lunak yang akan dibangun. Lalu kemudian dibuat program dalam bentuk *prototype* yang merupakan program yang belum jadi agar *user* mendapatkan gambaran tentang program yang dibangun (Shalahuddin & S, 2013).

## 2.9 Database Management System (DBMS)

*Database Management System* (DBMS) merupakan kumpulan program yang digunakan untuk mendefinisikan, mengatur, serta memproses *database*. *Database* memiliki esensi yaitu sebuah struktur yang dibangun untuk keperluan penyimpanan data. DBMS sendiri merupakan sebuah *tools* yang berguna untuk membangun struktur tersebut (Raharjo, 2011).

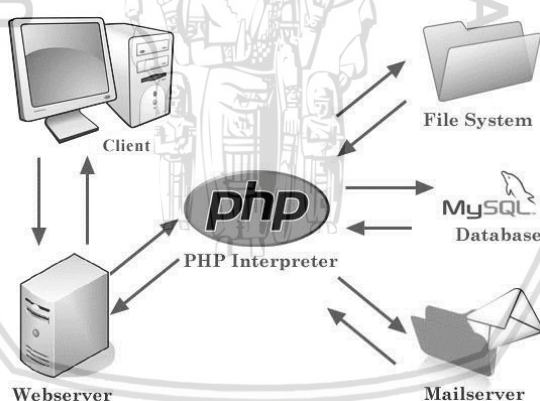
Dalam penelitian ini, *Database Management System* (DBMS) yang digunakan adalah DBMS versi *open source* yaitu MySQL. MySQL merupakan salah satu *database server* yang bersifat fleksibel. Pada MySQL, *user* dapat menyimpan informasi dalam jumlah yang banyak ataupun sedikit. MySQL menggunakan perintah dasar *Structured Query Language* (SQL) yang merupakan bahasa yang biasa digunakan untuk mengelola data pada *Database Management System*.



## 2.10 PHP

PHP merupakan salah satu bahasa pemrograman dan mempunyai kepanjangan yaitu *PHP Hypertext Preprocessor*. PHP merupakan skrip yang bersifat *server site* dimana proses pengerjaan *script*nya berlangsung di *server*. Pertama kali dikembangkan oleh salah seorang *programmer* bernama Rasmus Lerdorf. Awalnya, PHP merupakan singkatan dari *Personal Home Page Tools*, setelah dikembangkan oleh Andi Gutmans dan Zeev Suraski, fitur PHP bertambah dan kemudian diubah singkatannya menjadi *PHP Hypertext Preprocessor*. Salah satu kelebihan PHP adalah kemudahannya untuk berinteraksi dengan *database*. PHP juga bisa mendukung *database* secara langsung tanpa harus menginstall *connector*. PHP sangatlah fleksibel untuk berhubungan dengan berbagai *database*. Dengan menggunakan PHP maka perawatan *web site* akan lebih mudah. Penulisan *script* PHP disisipkan pada HTML (Sulhan, 2007).

Cara kerja PHP ditunjukkan pada Gambar 2.3 berikut, cara kerja PHP diawali dengan permintaan suatu halaman *web* yang dilakukan oleh *browser* atau *client*. Kemudian berdasarkan alamat pada URL, *browser* mendapatkan alamat dari *webserver*, yang kemudian mengidentifikasi halaman yang diminta, dan menyampaikan segala informasi yang dibutuhkan oleh *webserver*. Selanjutnya, ketika file PHP yang diminta didapatkan oleh *webserver*, isi dari file PHP tersebut segera dikirimkan ke mesin PHP untuk diproses dan hasilnya berupa kode HTML ke *webserver*, lalu kemudian menyimpannya ke *client*.



**Gambar 2.3 Cara Kerja PHP**

(EMS, 2016)

## 2.11 XAMPP

XAMPP merupakan sebuah *software web server* apache yang menyediakan database server MySQL dan mendukung pemrograman dengan bahasa PHP. XAMPP merupakan perangkat lunak yang dapat digunakan secara gratis dan dapat diinstall pada sistem operasi linux dan windows (Puspitasari, 2011).

XAMPP juga berperan sebagai *server web* pada sebuah komputer. XAMPP juga dapat dikatakan sebagai CPanel *server virtual* yang dapat membantu *developer*

melakukan *preview* sehingga *developer* dapat memodifikasi *website* tanpa harus terhubung dengan internet.

## 2.12 CodeIgniter

CodeIgniter merupakan salah satu *framework web* yang digunakan untuk pemrograman bahasa PHP. CodeIgniter memiliki banyak fitur yang dapat membantu *developer* PHP untuk dapat mengembangkan aplikasi web dengan mudah dan cepat (Raharjo, 2018). CodeIgniter memiliki desain yang lebih sederhana serta fleksibel.

Codeigniter menggunakan arsitektur atau pola perancangan MVC. MVC adalah sebuah komponen aplikasi yang berupa kelas, dimana kelas merupakan inti dari gaya pemrograman berorientasi objek.

## 2.13 HTML

*Hypertext Markup Language* merupakan sebuah bahasa pemrograman yang sering digunakan untuk membangun atau membuat sebuah halaman web. HTML merupakan penyusun utama dari antarmuka (*user interface*) sebuah web. Bahasa pemrograman ini sangat mudah untuk dipelajari dan dipahami. HTML terdiri dari tag serta aturan-aturan yang memungkinkan *user* untuk membuat dokumen *hypertext*. HTML menambahkan dokumen dengan *table* yang menandakan bagaimana teks di dokumen harus disajikan dan bagaimana agar dokumen bias saling dihubungkan. Pada skema tambahan HTML terdapat kekuatan untuk membuat berbagai aplikasi *client-server*, *form*, *multimedia*, dan halaman interaktif. HTML adalah dokumen ASCII atau teks biasa, yang dirancang agar tidak tergantung pada sebuah sistem operasi tertentu (EMS, 2016).

## 2.14 Pendekatan Berorientasi Objek

Pendekatan berorientasi objek adalah sebuah teknik pendekatan untuk melihat permasalahan serta sistem, dalam hal ini sistem yang dimaksud adalah sistem informasi, sistem perangkat lunak, dan sistem lainnya. PBO akan melihat sebuah sistem yang dibangun sebagai kumpulan objek-objek yang memiliki koresponden dengan objek pada dunia nyata.

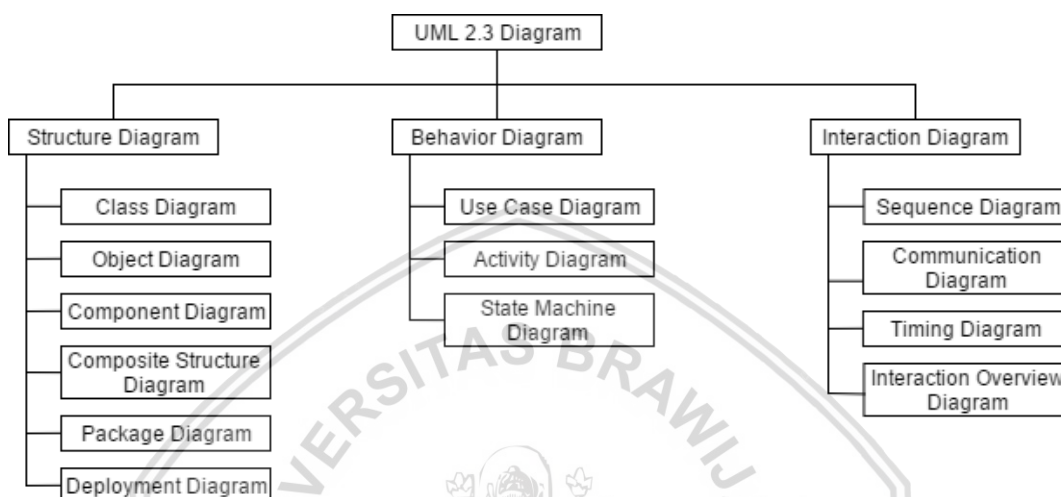
Pada rekayasa perangkat lunak, konsep PBO bisa diterapkan pada tahap analisis, perancangan, implementasi, serta pengujian sistem. Terdapat berbagai teknik yang bisa digunakan pada setiap tahap yang telah disebutkan, dengan menggunakan aturan serta alat bantu pemodelan tertentu (Shalahuddin & S, 2013).

## 2.15 Unified Model Language

UML adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram serta beberapa teks pendukung. UML berfungsi untuk melakukan pemodelan saja (Shalahuddin & S, 2013).

UML digunakan untuk memvisualisasikan, menspesifikasikan, memodifikasi, membangun, serta mendokumentasikan alat dan juga ide-ide yang ada pada pengembangan perangkat lunak berbasis objek (S, 2012).

Gambar 2.4 berikut menunjukkan bahwa diagram UML terbagi atas 3 jenis, yaitu *structure diagram*, *behaviour diagram*, dan *interaction diagram*. Pada penelitian ini, diagram UML yang digunakan adalah *class diagram*, *use case diagram*, dan *sequence diagram*.



**Gambar 2.4 Jenis Diagram UML**  
(Shalahuddin & S, 2013)

### 2.15.1 Class Diagram

Diagram kelas atau *class diagram* adalah gambaran struktur sistem yang didefinisikan oleh kelas-kelas yang akan dibuat dan digunakan untuk membangun sebuah sistem. Sebuah kelas memiliki bagian sebagai berikut :

#### 2.15.1.1 Atribut


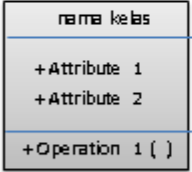






Atribut merupakan sebuah variable yang dimiliki oleh suatu kelas.

#### 2.15.1.2 Method

Method atau biasa juga disebut operasi merupakan sebuah fungsi yang dimiliki oleh suatu kelas.

Diagram kelas dibuat untuk memudahkan *programmer* atau pengembang untuk membuat kelas-kelas sesuai dengan rancangan pada diagram kelas agar dokumentasi perangkat lunak bisa sesuai atau sinkron (Shalahuddin & S, 2013). Pada Tabel 2.1 berikut menunjukkan simbol-simbol yang biasa digunakan pada diagram kelas yaitu *package*, operasi, antarmuka, serta relasi antar kelas.

Tabel 2.1 Simbol-simbol pada *Class Diagram*

Simbol	Deskripsi
	Package merupakan sebuah bungkus dari satu atau lebih kelas
	Kelas pada struktur sistem
	sama dengan konsep interface dalam pemrograman berorientasi objek
	relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
	relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity
	relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
	relasi antar kelas dengan makna kebergantungan antar kelas
	relasi antar kelas dengan makna semua-bagian (whole-part)


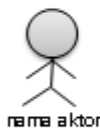





(Shalahuddin & S, 2013)

### 2.15.2 Use Case Diagram

Use case diagram merupakan pemodelan untuk *behavior* atau perilaku sistem yang akan dibuat. Use case diagram mendeskripsikan sebuah interaksi yang terjadi antara satu *actor* atau lebih dengan sistem yang akan dibuat. Use case diagram digunakan untuk mengetahui fungsi apa saja yang terdapat didalam sebuah sistem dan siapa *actor* yang berhak menggunakan fungsi-fungsi tersebut. Adapun syarat untuk memberikan nama pada sebuah use case harus didefinisikan sesederhana mungkin agar mudah dipahami (Shalahuddin & S, 2013). Pada Tabel 2.2 berikut

menunjukkan simbol-simbol yang biasa digunakan pada *use case diagram* yaitu nama *use case*, aktor, relasi antar *use case*.

**Tabel 2.2 Simbol-simbol pada Use Case Diagram**

Simbol	Deskripsi
<p>Use Case</p> 	<p>fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit dan aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case</p>
<p>Aktor</p> 	<p>orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor</p>
<p>Asosiasi / association</p> 	<p>komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor</p>
<p>Extend</p> 	<p>relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walaupun tanpa use case tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek; biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, arah panah menunjukan pada use case yang dituju contoh :</p> 
<p>Include</p> 	<p>relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini. Ada dua sudut pandang yang cukup besar mengenai include di use case, include berarti use case yang ditambahkan akan selalu dipanggil saat use case tambahan dijalankan, contoh :</p> 



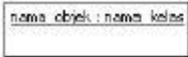

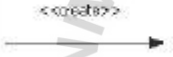
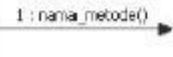
(Shalahuddin & S, 2013)

### 2.15.3 Sequence Diagram

Diagram sekuen atau *sequence diagram* merupakan gambaran dari *behavior* atau perilaku sebuah objek pada use case dengan mendeskripsikan *lifetime* (waktu hidup) dan *message* yang dikirimkan dan diterima antar objek. Untuk membuat *sequence diagram* maka terlebih dulu harus diketahui objek apa saja yang terlibat dalam sebuah use case beserta *method* yang dimiliki (Shalahuddin & S, 2013). Pada Tabel 2.3 berikut menunjukkan simbol-simbol yang biasa digunakan pada *sequence diagram* yaitu nama aktor, *lifeline*, objek, dan *message*.



Tabel 2.3 Simbol-simbol pada *Sequence Diagram*

Simbol	Deskripsi
atau  tanpa waktu aktif	orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
Garis hidup / <i>lifeline</i> 	menyatakan kehidupan suatu objek
Objek 	menyatakan objek yang berinteraksi pesan
Waktu aktif 	menyatakan objek dalam keadaan aktif dan berinteraksi pesan
Pesan tipe create 	menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
Pesan tipe call 	menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,

(Shalahuddin & S, 2013)

## 2.16 Pengujian Perangkat Lunak

Pada perangkat lunak sering terjadi kesalahan atau *error* atau *bug* pada beberapa proses tertentu saat perangkat lunak sudah di kirim ke pengguna. Untuk menghindari adanya *bug* tersebut, diperlukan pengujian perangkat lunak sebelum perangkat lunak dikirim ke pengguna atau selama proses pengembangan perangkat lunak tersebut. Pengujian diperlukan tidak hanya untuk meminimalisir kesalahan secara teknis tetapi juga secara non teknis (Shalahuddin & S, 2013).

### 2.16.1 *White-Box Testing*

*White Box Testing* atau yang biasa disebut dengan pengujian kotak putih merupakan pengujian perangkat lunak dari segi desain dan juga kode program. *White Box Testing* dilakukan dengan memeriksa *logic* dari kode program. Pengujian terhadap dokumentasi yang dibuat juga harus dilakukan agar dokumentasi yang dibuat tetap konsisten dengan perangkat lunak yang dibuat (Shalahuddin & S, 2013).



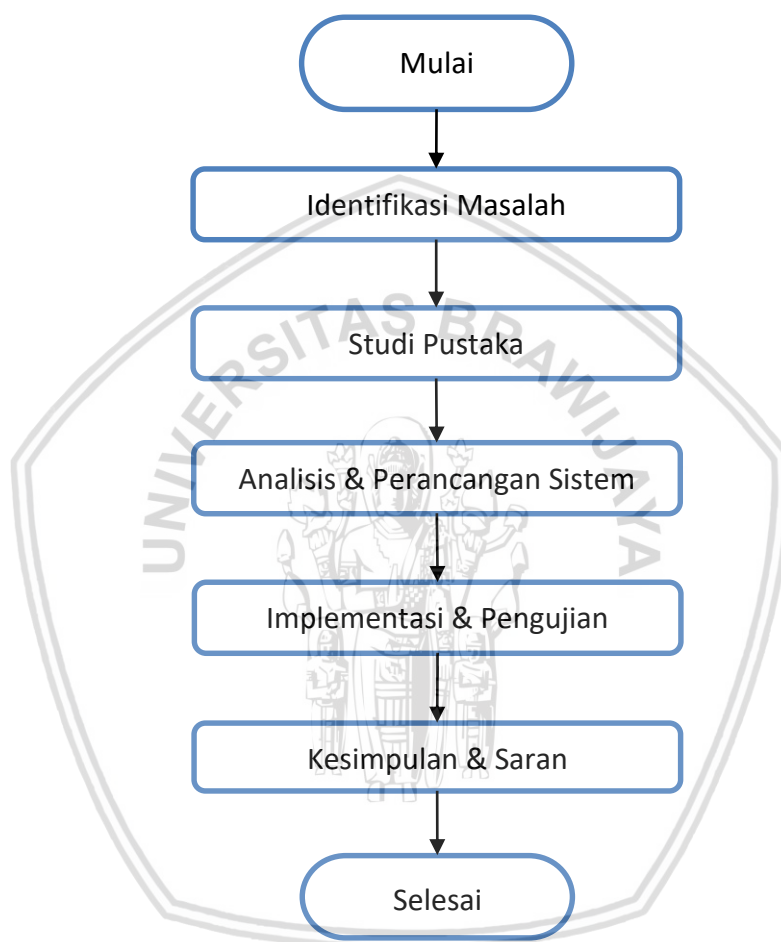
### 2.16.2 Black-Box Testing

*Black Box Testing* atau yang biasa disebut dengan pengujian kotak hitam merupakan pengujian perangkat lunak dari segi fungsionalitas tanpa melihat desain dan juga kode programnya. Pengujian ini bertujuan untuk mengetahui apakah perangkat lunak sudah sesuai dengan fungsi-fungsi, *input*, dan juga *output* yang dibutuhkan (Shalahuddin & S, 2013).



## BAB 3 METODOLOGI

Dalam melakukan penelitian tugas akhir atau skripsi ini, metode penelitian merupakan sebuah pedoman yang dapat digunakan dalam pelaksanaan penelitian. Dalam metodologi penelitian ini, akan dijelaskan langkah demi langkah penelitian yang akan dilakukan. Tahap-tahap penelitian akan ditunjukkan pada Gambar 3.1 berikut.



**Gambar 3.1 Diagram Alur Metodologi Penelitian**

### 3.1 Identifikasi Masalah

Dalam mendapatkan masalah penelitian, berpikir merupakan kegiatan pertama yang harus dilakukan untuk dapat memperoleh ide atau gagasan serta motivasi untuk bisa melakukan sebuah penelitian. Setelah itu, dilakukan sebuah survey mengenai subjek serta objek dari masalah yang terjadi, lalu kemudian dilakukan identifikasi mengenai masalah yang ada. Masalah yang telah diidentifikasi kemudian dirumuskan dalam sebuah kalimat yang bias merepresentasikan serta mendeskripsikan penelitian yang akan dikerjakan. Rumusan masalah dituliskan

dalam bentuk penggabungan antara focus permasalahan dengan beberapa teori yang berkaitan.

### 3.2 Studi Pustaka

Setelah identifikasi masalah, tahap berikutnya yaitu mengumpulkan berbagai informasi mengenai ilmu-ilmu yang dapat digunakan untuk menyelesaikan permasalahan pada suatu penelitian. Studi pustaka merupakan metode yang berfokus pada pencarian penelitian yang berhubungan dan juga informasi dari berbagai dokumen, baik dalam bentuk tulisan, gambar, maupun dokumen elektronik yang dapat mendukung proses penelitian dan penulisan. Adapun studi pustaka yang terkait dengan penelitian ini adalah sebagai berikut:

1. Rekayasa Perangkat Lunak
2. Sistem Informasi Kesehatan
3. Rekam Medis
4. RSIA. Prof. Dr. H. M. Farid Makassar
5. Web
6. *Software Development Life Cycle (SDLC)*
7. *Database Management System (DBMS)*
8. PHP
9. XAMPP
10. CodeIgniter
11. HTML
12. Pemrograman Berorientasi Objek
13. Pengujian

### 3.3 Analisis Kebutuhan

Adapun analisis kebutuhan yang bertujuan untuk mendapatkan segala kebutuhan yang dibutuhkan dalam membangun suatu sistem. Kebutuhan tersebut merupakan kebutuhan *user*/pengguna yang dimodelkan ke dalam bentuk *use case diagram* dan *use case scenario*. Adapun kebutuhan yang akan digunakan dalam implementasi aplikasi rekam medis ini adalah sebagai berikut :

#### 3.3.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang berhubungan dengan fungsi atau apa yang dilakukan oleh sistem, bagaimana cara kerja dari sebuah sistem, bagaimana sistem memberikan reaksi pada sebuah *input*/masukan tertentu, serta bagaimana perilaku sistem pada situasi tertentu.

### 3.3.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional merupakan kebutuhan yang digunakan sebagai penunjang agar dapat memudahkan *user/pengguna* saat menggunakan sistem.

### 3.4 Perancangan Sistem

Pada penelitian ini, perancangan sistem dilakukan dengan menggunakan *Object Oriented Programming* (OOP) dalam bentuk *Sequence Diagram*, *Class Diagram*, perancangan *database* MySQL, serta perancangan *user interface*.

### 3.5 Implementasi Sistem

Implementasi dilakukan sesuai dengan analisis kebutuhan serta perancangan sistem yang telah dilakukan sebelumnya sesuai dengan diagram alur penelitian. Implementasi sistem dilakukan setelah proses perancangan sistem selesai menggunakan bahasa pemrograman PHP serta database MySQL.

### 3.6 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa sistem yang telah dibuat bekerja sesuai dengan apa yang diinginkan oleh *user/pengguna* dan sesuai dengan spesifikasi kebutuhan. Tahap ini juga berguna untuk mengetahui kelemahan dan juga error/kesalahan yang terjadi pada sistem yang telah dibuat. Pengujian yang akan dilakukan pengujian fungsional dan non fungsional. Pengujian yang akan digunakan adalah pengujian *white box* dan *black box*.

### 3.7 Kesimpulan & Saran

Tahap akhir dari penelitian ini adalah pengambilan kesimpulan dan saran. Kesimpulan diadaptasi dari hasil pengujian serta analisa yang telah dilakukan. Penulisan saran juga dilakukan untuk pengembangan lebih lanjut serta untuk memperbaiki kesalahan-kesalahan yang ada.

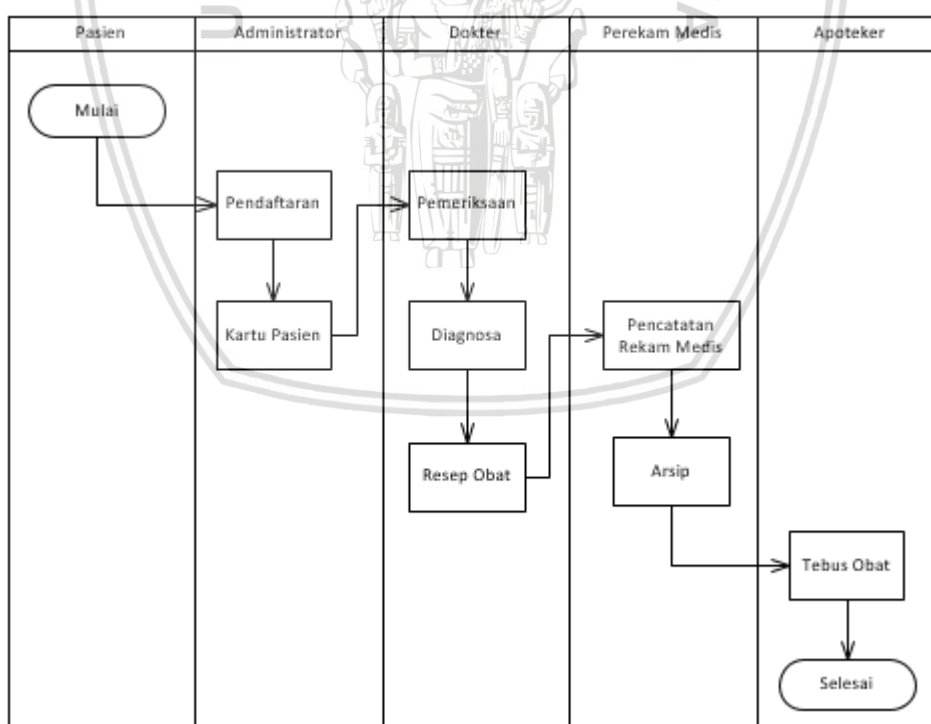
## BAB 4 ANALISIS KEBUTUHAN SISTEM

Pada bab ini akan dibahas mengenai analisis permasalahan dan analisis kebutuhan sistem pada RSIA. Prof. H. M. Farid Makassar. Analisis permasalahan serta kebutuhan sistem dibuat berdasarkan pengumpulan data, dengan cara wawancara dan observasi lapangan terhadap *stakeholder* yang berkaitan dengan sistem yang akan digunakan pada RSIA. Prof. H. M. Farid Makassar.

### 4.1 Analisis Permasalahan

Berdasarkan pengumpulan data dengan cara wawancara serta observasi lapangan, pengolahan data pada RSIA. Prof. H. M. Farid Makassar masih dilakukan dengan cara konvensional dalam pencatatan data rekam medik pasien dibutuhkan waktu yang cukup lama dalam pencatatan, penyampaian informasi data rekam medik memerlukan waktu yang cukup lama karena data yang dicari cukup banyak, belum adanya sistem keamanan data sehingga orang yang tidak memiliki wewenang dapat menyalahgunakan data, serta untuk pelaporan informasi jumlah pasien per bulan kepada kepala RSIA masih memerlukan waktu yang lama karena user harus melakukan pencarian data terlebih dulu pada lemari arsip dan melakukan rekapitulasi secara manual.

#### 4.1.1 Proses Bisnis pada RSIA. Prof. Dr. H. M. Farid Makassar



Gambar 4.1 Diagram Proses Bisnis Rumah Sakit

(Dika, 2014)

#### 1. Pendaftaran

Ini merupakan tahap awal saat pasien datang, hal pertama yang harus dilakukan pasien adalah mengisi dan melengkapi formulir pendaftaran pasien di bagian administrasi dan akan mendapatkan kartu pasien dan arsip berupa rekam medis.

#### 2. Poli Pemeriksaan

Setelah melakukan pendaftaran, pasien akan diarahkan ke bagian poli untuk mendapatkan pelayanan dan tindakan dokter. Pada proses ini, pasien akan diberikan penjelasan mengenai kondisi pasien dan juga resep obat.

#### 3. Pencatatan Rekam Medis

Setelah dilakukan pemeriksaan, hasil pemeriksaan berupa diagnosa, tindakan akan dimasukkan ke dalam rekam medis guna menjadi histori pemeriksaan pasien.

#### 4. Tebus Obat

Pada tahap ini, pasien menebus obat yang telah diberikan oleh dokter. Apabila obat yang tertulis didalam resep habis atau tidak ada di apotek rumah sakit, maka pasien bisa mendatangi apotek lain dengan membawa serta resep obat tersebut.

### 4.1.2 Dokumen Rekam Medis RSIA. Prof. Dr. H. M. Farid Makassar

Dokumen rekam medis pada RSIA. Prof. Dr. H. M. Farid Makassar terbagi atas 2 jenis, yaitu:

1. Map hijau yang ditujukan untuk pasien seorang ibu,
2. Map kuning yang ditujukan untuk pasien bayi baru lahir,
3. Reporting yang berguna sebagai pelaporan untuk dinas kesehatan mengenai semua kegiatan penyelenggaraan rumah sakit.

### 4.2 Gambaran Umum Aplikasi

Aplikasi yang akan dibangun pada penelitian ini adalah sebuah aplikasi rekam medis yang sekaligus merupakan sistem informasi pendataan pasien. Informasi yang terdapat dalam aplikasi ini hanya berupa informasi berupa data pribadi pasien dimana pasien tersebut adalah seorang ibu, data rekam medis pasien baik rawat inap maupun rawat jalan. Dengan adanya penelitian ini, diharapkan dapat mempermudah proses rekapitulasi data pasien, dan pelaporan kegiatan pada RSIA. Prof. Dr. H. M. Farid Makassar.

### 4.3 Identifikasi Aktor

Ini merupakan tahap identifikasi aktor-aktor yang akan berinteraksi dengan aplikasi. Pada Tabel 4.1 akan dijelaskan aktor-aktor yang terlibat.



Tabel 4.1 Identifikasi Aktor

No	Aktor	Deskripsi
1	Perekam Medis	Pengguna yang bertugas untuk melakukan pendaftaran pasien, melihat dan mencari data rekam medis pasien, melakukan <i>reporting</i> . Petugas medis merupakan pihak yang mengisi kelengkapan berkas pasien, dan melakukan rekapitulasi data rekam medis pasien ( <i>reporting</i> ).
2	Perawat	Pengguna yang bertugas untuk melakukan <i>input</i> data dari setiap <i>visite</i> yang dilakukan oleh dokter.
3	Guest	Pengguna yang belum masuk ke dalam sistem.

#### 4.4 Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan utama yang harus dimiliki oleh sistem. Dan merupakan kebutuhan yang menjadi penunjang sistem tersebut sesuai dengan tujuannya. Setiap kebutuhan akan diberikan kode fungsi RM\_FAABB, dimana RM merupakan nama aplikasi yang akan dibuat yaitu Rekam Medis, F merupakan kode yang merepresentasikan kebutuhan fungsional, AA merupakan nomor kebutuhan utama dan BB merupakan kode spesifikasi kebutuhan utama. Spesifikasi kebutuhan fungsional ditunjukkan pada Tabel 4.2 :

Tabel 4.2 Spesifikasi Kebutuhan Fungsional dan Pemetaan Use Case

No	Kode	Deskripsi/Spesifikasi	Use Case	Aktor
1	RM_F0100	Sistem dapat menyeleksi pengguna dengan melakukan validasi terhadap user untuk mengakses sistem sesuai dengan otoritasnya.	Login	Guest
2	RM_F0200	Sistem dapat menghapus <i>session</i> pengguna yang sudah masuk ke dalam sistem.	Logout	Perekam Medis
3	RM_F0300	Sistem harus mampu menampilkan daftar data pasien.	Lihat Daftar Data Pasien	Perekam Medis
3.1	RM_F0301	Sistem menyediakan informasi data pasien seperti No RM, nama, tanggal lahir, alamat, telp,		

		alergi dan keluhan dalam bentuk tabel.		
4	RM_F0400	Sistem harus mampu menambah data pasien yang baru.	Tambah Data Pendaftaran Pasien	Perekam Medis
4.1	RM_F0401	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data pasien yang berisi No RM, tgl lahir, nama, jenis kelamin, agama, pekerjaan, alamat, telp, alergi, dan keluhan.		
5	RM_F0500	Sistem harus mampu mengedit data pasien.	Edit Data Pendaftaran Pasien	Perekam Medis
5.1	RM_F0501	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data pasien.		
6	RM_F0600	Sistem harus mampu mencari data pasien.	Cari Data Pasien	Perekam Medis
6.1	RM_F0601	Sistem menyediakan tombol cari untuk mencari data pasien dan menampilkan data pasien yang dicari.		
7	RM_F0700	Sistem harus mampu menambah data ringkasan masuk dan keluar pasien.	Tambah Data Ringkasan Masuk dan Keluar	Perekam Medis
7.1	RM_F0701	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data ringkasan masuk dan keluar.		
8	RM_F0800	Sistem harus mampu menampilkan daftar data ringkasan masuk dan keluar pasien	Lihat Data Ringkasan Masuk dan Keluar	Perekam Medis

8.1	RM_F0801	Sistem menyediakan informasi data ringkasan masuk dan keluar seperti tanggal, nama, dan diagnose dalam bentuk tabel.		
9	RM_F0900	Sistem harus mampu menampilkan detail data ringkasan masuk dan keluar pasien.	Lihat Detail Data Ringkasan Masuk dan Keluar	Perekam Medis
9.1	RM_F0901	Sistem menyediakan tombol untuk melihat detail data ringkasan masuk dan keluar pasien.		
10	RM_F1000	Sistem harus mampu mengedit data ringkasan masuk dan keluar pasien.	Edit Data Ringkasan Masuk dan Keluar	Perekam Medis
10.1	RM_F1001	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data ringkasan dan keluar masuk pasien.		
11	RM_F1100	Sistem harus mampu mencari data ringkasan masuk dan keluar pasien.	Cari Data Ringkasan Masuk dan Keluar	Perekam Medis
11.1	RM_F1101	Sistem menyediakan tombol cari untuk mencari data ringkasan masuk dan keluar pasien dan menampilkan data pasien yang dicari.		
12	RM_F1200	Sistem harus mampu menambah data resume medis pasien.	Tambah Data Resume Medis	Perekam Medis
12.1	RM_F1201	Sistem menyediakan tombol tambah data dan <i>form</i> untuk		

		menambahkan data resume medis.		
13	RM_F1300	Sistem harus mampu menampilkan daftar data resume medis pasien	Lihat Data Resume Medis	Perekam Medis
13.1	RM_F1301	Sistem menyediakan informasi data resume medis seperti tanggal, nama, dan diagnose dalam bentuk tabel.		
14	RM_F1400	Sistem harus mampu menampilkan detail data resume medis pasien.	Lihat Detail Data Resume Medis	Perekam Medis
14.1	RM_F1401	Sistem menyediakan tombol untuk melihat detail data resume medis pasien.		
15	RM_F1500	Sistem harus mampu mengedit data resume medis pasien.	Edit Data Resume Medis	Perekam Medis
15.1	RM_F1501	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data resume medis pasien.		
16	RM_F1600	Sistem harus mampu mencari data resume medis pasien.	Cari Data Resume Medis	Perekam Medis
16.1	RM_F1601	Sistem menyediakan tombol cari untuk mencari data resume medis pasien dan menampilkan data pasien yang dicari.		
17	RM_F1700	Sistem harus mampu menambah data rencana pulang pasien.	Tambah Data Rencana Pulang	Perekam Medis
17.1	RM_F1701	Sistem menyediakan tombol tambah data		

		dan <i>form</i> untuk menambahkan data rencana pulang.		
18	RM_F1800	Sistem harus mampu menampilkan daftar data rencana pulang pasien	Lihat Data Rencana Pulang	Perekam Medis
18.1	RM_F1801	Sistem menyediakan informasi data bayi seperti tanggal, nama, dan diagnose akhir dalam bentuk tabel.		
19	RM_F1900	Sistem harus mampu menampilkan detail data rencana pulang pasien.	Lihat Detail Data Rencana Pulang	Perekam Medis
19.1	RM_F1901	Sistem menyediakan tombol untuk melihat detail data rencana pulang pasien.		
20	RM_F2000	Sistem harus mampu mengedit data rencana pulang pasien.	Edit Data Rencana Pulang	Perekam Medis
20.1	RM_F2001	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data rencana pulang pasien.		
21	RM_F2100	Sistem harus mampu mencari data rencana pulang pasien.	Cari Data Rencana Pulang	Perekam Medis
21.1	RM_F2101	Sistem menyediakan tombol cari untuk mencari data rencana pulang pasien dan menampilkan data pasien yang dicari.		
22	RM_F2200	Sistem harus mampu menambah data	Tambah Data Pengkajian	Perekam Medis

		pengkajian asuhan kebidanan pasien.	Asuhan Kebidanan	
22.1	RM_F2201	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data pengkajian asuhan kebidanan.		
23	RM_F2300	Sistem harus mampu menampilkan daftar data pengkajian asuhan kebidanan pasien	Lihat Data Pengkajian Asuhan Kebidanan	Perekam Medis
23.1	RM_F2301	Sistem menyediakan informasi data pengkajian asuhan kebidanan seperti tanggal, nama, dan diagnose dalam bentuk tabel.		
24	RM_F2400	Sistem harus mampu menampilkan detail data pengkajian asuhan kebidanan pasien.	Lihat Detail Data Pengkajian Asuhan Kebidanan	Perekam Medis
24.1	RM_F2401	Sistem menyediakan tombol untuk melihat detail data pengkajian asuhan kebidanan pasien.		
25	RM_F2500	Sistem harus mampu mengedit data pengkajian asuhan kebidanan pasien.	Edit Data Pengkajian Asuhan Kebidanan	Perekam Medis
25.1	RM_F2501	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data pengkajian asuhan kebidanan pasien.		
26	RM_F2600	Sistem harus mampu mencari data pengkajian asuhan kebidanan pasien.	Cari Data Pengkajian Asuhan Kebidanan	Perekam Medis
26.1	RM_F2601	Sistem menyediakan tombol cari untuk		



		mencari data pengkajian asuhan kebidanan pasien dan menampilkan data pasien yang dicari.		
27	RM_F2700	Sistem harus mampu menambah data pengkajian asuhan keperawatan pasien.	Tambah Data Pengkajian Asuhan Keperawatan	Perekam Medis
27.1	RM_F2701	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data pengkajian asuhan keperawatan.		
28	RM_F2800	Sistem harus mampu menampilkan daftar data pengkajian asuhan keperawatan pasien	Lihat Data Pengkajian Asuhan Keperawatan	Perekam Medis
28.1	RM_F2801	Sistem menyediakan informasi data pengkajian asuhan keperawatan seperti tanggal, nama, dan diagnose dalam bentuk tabel.		
29	RM_F2900	Sistem harus mampu menampilkan detail data pengkajian asuhan keperawatan pasien.	Lihat Detail Data Pengkajian Asuhan Keperawatan	Perekam Medis
29.1	RM_F2001	Sistem menyediakan tombol untuk melihat detail data pengkajian asuhan keperawatan pasien.		
30	RM_F3000	Sistem harus mampu mengedit data pengkajian asuhan keperawatan pasien.	Edit Data Pengkajian Asuhan Keperawatan	Perekam Medis
30.1	RM_F3001	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data		

		pengkajian asuhan keperawatan pasien.		
31	RM_F3100	Sistem harus mampu mencari data pengkajian asuhan keperawatan pasien.	Cari Data Pengkajian Asuhan Keperawatan	Perekam Medis
31.1	RM_F3101	Sistem menyediakan tombol cari untuk mencari data pengkajian asuhan keperawatan pasien dan menampilkan data pasien yang dicari.		
32	RM_F3200	Sistem harus mampu menambah data MR3.1 pasien.	Tambah Data MR3.1	Perekam Medis
32.1	RM_F3201	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data MR3.1.		
33	RM_F3300	Sistem harus mampu menampilkan daftar data MR3.1 pasien	Lihat Data MR3.1	Perekam Medis
33.1	RM_F3301	Sistem menyediakan informasi data MR3.1 seperti tanggal, nama, dan no rm bayi dalam bentuk tabel.		
34	RM_F3400	Sistem harus mampu menampilkan detail data MR3.1 pasien.	Lihat Detail Data	Perekam Medis
34.1	RM_F3401	Sistem menyediakan tombol untuk melihat detail data MR3.1 pasien.		
35	RM_F3500	Sistem harus mampu mengedit data MR3.1 pasien.	Edit Data MR3.1	Perekam Medis
35.1	RM_F3501	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data MR3.1 pasien.		

36	RM_F3600	Sistem harus mampu mencari data MR3.1 pasien.	Cari Data MR3.1	Perekam Medis
36.1	RM_F3601	Sistem menyediakan tombol cari untuk mencari data MR3.1 pasien dan menampilkan data pasien yang dicari.		
37	RM_F3700	Sistem harus mampu menambah data MR3.2 pasien.	Tambah Data MR3.2	Perekam Medis
37.1	RM_F3701	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data MR3.2.		
38	RM_F3800	Sistem harus mampu menampilkan daftar data MR3.2 pasien	Lihat Data MR3.2	Perekam Medis
38.1	RM_F3801	Sistem menyediakan informasi data MR3.2 seperti tanggal, nama, dan nama bayi dalam bentuk tabel.		
39	RM_F3900	Sistem harus mampu menampilkan detail data MR3.2 pasien.	Lihat Detail Data MR3.2	Perekam Medis
39.1	RM_F3901	Sistem menyediakan tombol untuk melihat detail data MR3.2 pasien.		
40	RM_F4000	Sistem harus mampu mengedit data MR3.2 pasien.	Edit Data MR3.2	Perekam Medis
40.1	RM_F4001	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data MR3.2 pasien.		
41	RM_F4100	Sistem harus mampu mencari data MR3.2 pasien.	Cari Data MR3.2	Perekam Medis
41.1	RM_F4101	Sistem menyediakan tombol cari untuk		

		mencari data MR3.2 pasien dan menampilkan data pasien yang dicari.		
42	RM_F4200	Sistem harus mampu menambah data rawat jalan pasien.	Tambah Data Rawat Jalan	Perekam Medis
42.1	RM_F4201	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data rawat jalan.		
43	RM_F4300	Sistem harus mampu menampilkan daftar data rawat jalan pasien	Lihat Data Rawat Jalan	Perekam Medis
43.1	RM_F4301	Sistem menyediakan informasi data rawat jalan seperti tanggal periksa, nama, dan keterangan dalam bentuk tabel.		
44	RM_F4400	Sistem harus mampu menampilkan detail data rawat jalan pasien.	Lihat Detail Rawat Jalan	Perekam Medis
44.1	RM_F4401	Sistem menyediakan tombol untuk melihat detail data rawat jalan pasien.		
45	RM_F4500	Sistem harus mampu mengedit data rawat jalan pasien.	Edit Data Rawat Jalan	Perekam Medis
45.1	RM_F4501	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data rawat jalan pasien.		
46	RM_F4600	Sistem harus mampu mencari data rawat jalan pasien.	Cari Data Rawat Jalan	Perekam Medis
46.1	RM_F4601	Sistem menyediakan tombol cari untuk mencari data rawat jalan pasien dan		

		menampilkan data pasien yang dicari.		
47	RM_F4700	Sistem harus mampu menambah data sebab kematian pasien.	Tambah Data Sebab Kematian	Perekam Medis
47.1	RM_F4701	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data sebab kematian.		
48	RM_F4800	Sistem harus mampu menampilkan detail data sebab kematian pasien.	Lihat Detail Sebab Kematian	Perekam Medis
49	RM_F4900	Sistem harus mampu melakukan cetak laporan laporan persalinan dan menampilkan <i>form</i> untuk memasukkan unit pelayanan, tanggal periode laporan dan nama pelapor.	Cetak Laporan Persalinan	Perekam Medis
49.1	RM_F4901	Sistem menampilkan preview laporan persalinan dalam bentuk pdf.		
50	RM_F5000	Sistem harus mampu melakukan cetak laporan laporan kematian dan menampilkan <i>form</i> untuk memasukkan unit pelayanan, tanggal periode laporan dan nama pelapor.	Cetak Laporan Kematian	Perekam Medis
50.1	RM_F5001	Sistem menampilkan preview laporan kematian dalam bentuk pdf.		

51	RM_F5100	Sistem harus mampu menambah data visite.	Tambah Data Visite	Perawat
51.1	RM_F5101	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data visite.		
52	RM_F5200	Sistem harus mampu menampilkan daftar data visite.	Lihat Data Visite	Perawat
52.1	RM_F5201	Sistem menyediakan informasi data visite seperti tanggal pemeriksaan, jam pemeriksaan, tindakan, dan obat dalam bentuk tabel.		
53	RM_F5300	Sistem harus mampu menampilkan detail data visite.	Lihat Detail Visite	Perawat
53.1	RM_F5301	Sistem menyediakan tombol untuk melihat detail data visite.		
54	RM_F5400	Sistem harus mampu mengedit data visite.	Edit Data Visite	Perawat
54.1	RM_F5401	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data visite.		
55	RM_F5500	Sistem harus mampu mencari data visite.	Cari Data Visite	Perawat
55.1	RM_F5501	Sistem menyediakan tombol cari untuk mencari data rawat jalan pasien dan menampilkan data visite yang dicari.		
56	RM_F5600	Sistem harus mampu menampilkan daftar data dokter.	Lihat Daftar Data Dokter	Perekam Medis
56.1	RM_F5601	Sistem menyediakan informasi data dokter seperti ID dokter, nama dokter,		



		alamat, no. telp, dan jadwal praktik dalam bentuk tabel.		
57	RM_F5700	Sistem harus mampu menambah data dokter yang baru.	Tambah Data Dokter	Perekam Medis
57.1	RM_F5701	Sistem menyediakan tombol tambah data dan <i>form</i> untuk menambahkan data pasien yang berisi ID dokter, nama dokter, alamat, no. telp, dan jadwal praktik.		
58	RM_F5800	Sistem harus mampu mengedit data dokter.	Edit Data Dokter	Perekam Medis
58.1	RM_F5801	Sistem menyediakan tombol edit dan <i>form</i> untuk mengedit data dokter.		
59	RM_F5900	Sistem harus mampu mencari data dokter.	Cari Data Dokter	Perekam Medis
59.1	RM_F5901	Sistem menyediakan tombol cari untuk mencari data dokter dan menampilkan data dokter yang dicari.		

#### 4.5 Kebutuhan Non Fungsional

Kebutuhan non fungsional merupakan kebutuhan yang mampu mendukung dan menunjang kebutuhan fungsional. Setiap kebutuhan akan diberikan kode fungsi RM\_NFAA, dimana RM merupakan nama aplikasi yang akan dibuat yaitu Rekam Medis, NF merupakan kode yang merepresentasikan kebutuhan non-fungsional, AA merupakan kode kebutuhan. Tabel 4.3 mendefinisikan kebutuhan non fungsional:

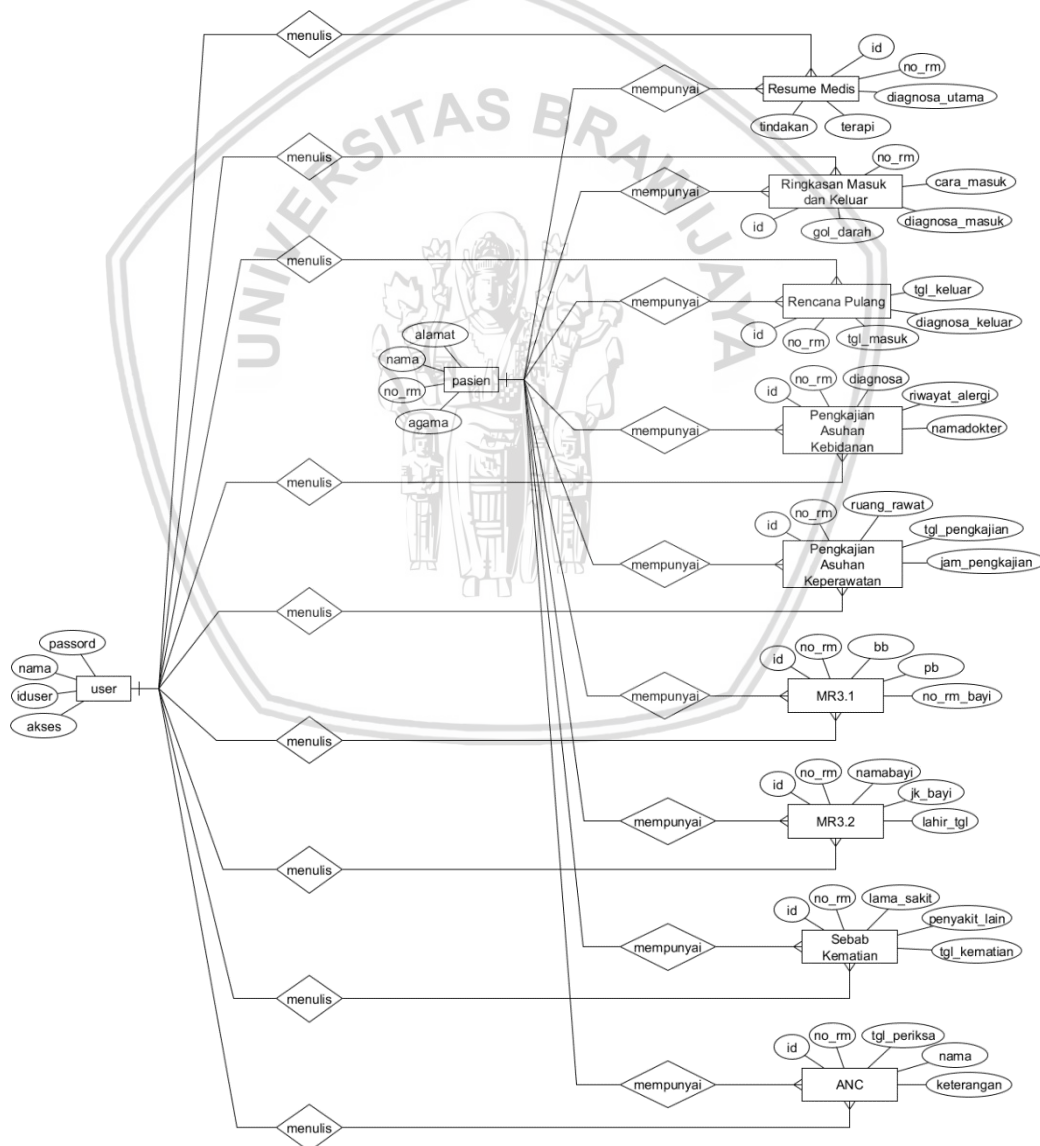
**Tabel 4.3 Kebutuhan Non Fungsional**

No	Kode Kebutuhan	Spesifikasi	Parameter
1	RM_NF01	Sistem harus dapat dijalankan di berbagai jenis browser seperti google chrome, MX5, internet explorer, dan firefox.	<i>Compatibility</i>

## 4.6 Analisis Data

Pada analisis data, menggunakan ERD (*Entity Relationship Diagram*) sebagai pemodelan awal dari basis data yang akan digunakan pada sistem dimana ERD sendiri terdiri atas entitas dan atribut.

Gambar 4.5 berikut merupakan diagram ERD Aplikasi Rekam Medis RSIAPF yang menunjukkan relasi antara entitas yang ada pada sistem. Dalam Aplikasi Rekam Medis RSIAPF terdapat 11 entitas yaitu user, pasien, ringkasan masuk dan keluar, resume medis, rencana pulang, pengkajian asuhan kebidanan, pengkajian asuhan keperawatan, mr3.1, mr3.2, rawat jalan dan sebab kematian. Setiap entitas yang ada pada Aplikasi Rekam Medis RSIAPF memiliki lebih dari 20 atribut sehingga pada *Entity Relationship Diagram* hanya digunakan 5 atribut saja untuk memudahkan pembacaan ERD.



**Gambar 4.2 Entity Relationship Diagram Aplikasi Rekam Medis RSIAPF**

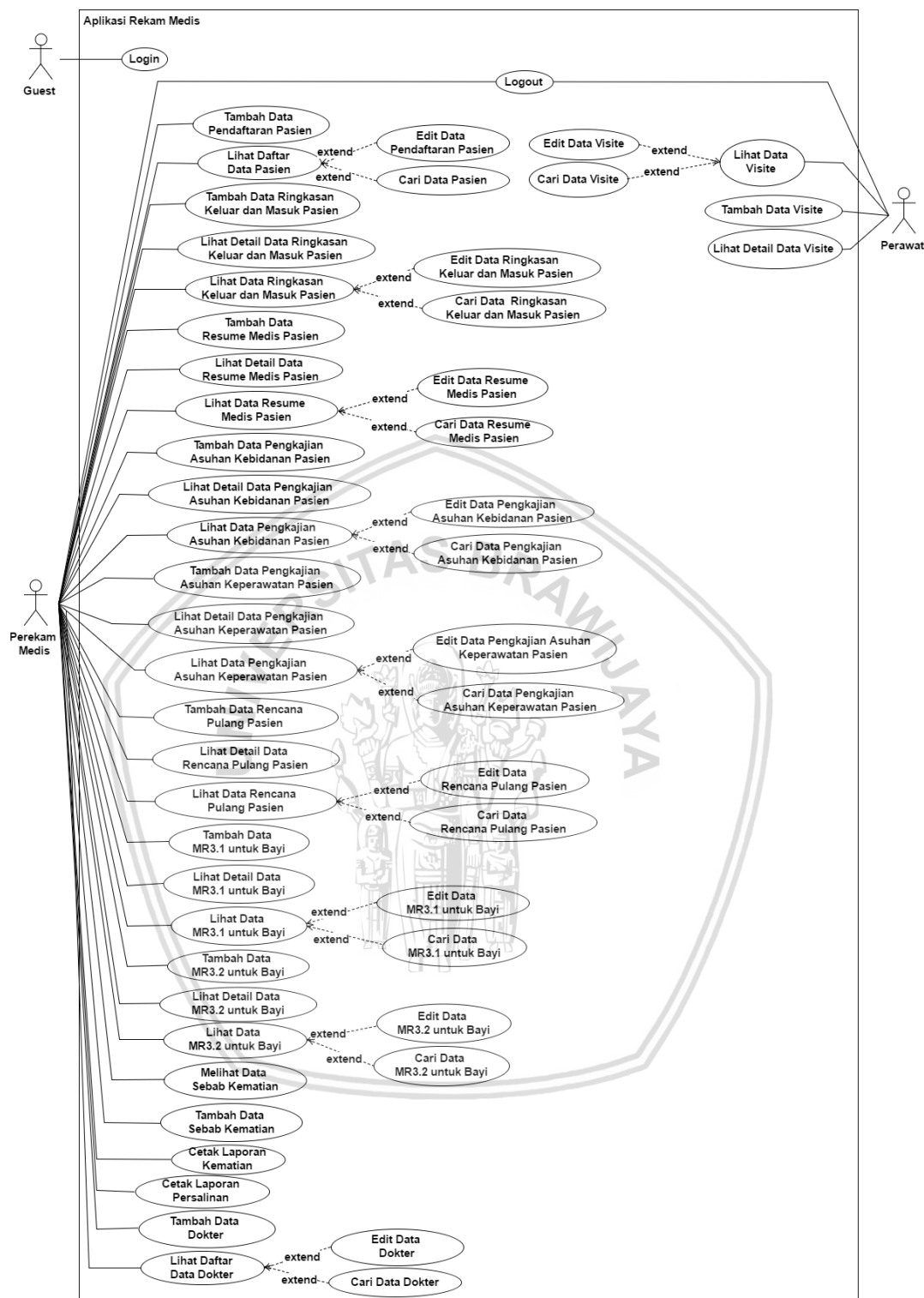
## 4.7 Pemodelan Kebutuhan Perangkat Lunak

Pemodelan kebutuhan perangkat lunak merupakan gambaran dari realita dan dituangkan dalam bentuk pemetaan dengan beberapa aturan tertentu. Pemodelan kebutuhan mencakup *use case diagram* dan *use case scenario*.

### 4.7.1 Use Case Diagram

*Use case diagram* digunakan untuk menjelaskan mengenai aktor yang terlibat dengan sistem yang dibangun beserta proses-proses yang ada didalamnya. Gambar 4.3 merupakan *use case diagram* yang dibuat berdasarkan analisis sistem yang sudah dijabarkan diatas dan dipatkan 59 kebutuhan fungsional dan 2 aktor.





Gambar 4.3 Use Case Diagram Aplikasi Rekam Medis RSIAPF

#### 4.7.2 Use Case Scenario

*Use case scenario* merupakan urutan proses yang dilakukan oleh sistem yang menghasilkan sesuatu yang dapat dilihat dan diamati oleh actor. *Use case scenario* merupakan *flow of event* atau alur proses dalam sebuah use case. Berikut ini merupakan alur proses dari setiap use case diatas:

Tabel 4.4 Use Case Scenario Login

Login	
Objective	Aktor dapat masuk ke dalam sistem sesuai dengan hak aksesnya
Actors	<i>Guest</i>
Pre condition	Aktor harus memiliki <i>username</i> dan <i>password</i> yang sudah terdaftar di <i>database</i> sistem
Main flow	<ol style="list-style-type: none"> <li>1. Aktor masuk ke halaman utama sistem</li> <li>2. Sistem menampilkan form <i>login</i></li> <li>3. Aktor memasukkan <i>username</i> dan <i>password</i></li> <li>4. Aktor menekan tombol <i>login</i></li> <li>5. Sistem melakukan validasi <i>username</i> dan <i>password</i> di <i>database</i></li> <li>6. Sistem akan menampilkan halaman beranda</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Tidak ada input pada <i>username</i> dan <i>password</i></li> <li>2. Sistem akan menampilkan pesan "<i>Please fill out this field</i>"</li> <li>3. Terjadi kesalahan input <i>username</i> atau <i>password</i></li> <li>4. Sistem akan menampilkan pesan "<i>Username/Password yang Anda masukkan salah</i>"</li> </ol>
Post condition	Aktor masuk ke dalam sistem

Tabel 4.5 Use Case Scenario Logout

Logout	
Objective	Aktor keluar dari sistem
Actors	Perekam Medis, Perawat
Pre condition	Aktor <i>login</i> di dalam sistem
Main flow	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol <i>logout</i></li> <li>2. Sistem menampilkan kembali halaman <i>login</i></li> </ol>
Alternative flow	-
Post condition	Aktor berhasil keluar dari sistem

Tabel 4.6 Use Case Scenario Tambah Data Pendaftaran Pasien

Tambah Data Pendaftaran Pasien	
Objective	Aktor menambahkan data pasien dengan mengisi <i>form</i> pendaftaran pasien
Actors	Perekam Medis
Pre condition	Aktor melihat daftar data pasien
Main flow	<ol style="list-style-type: none"> <li>1. Aktor masuk ke halaman beranda</li> <li>2. Sistem menampilkan halaman beranda</li> <li>3. Aktor menekan tombol data pasien</li> <li>4. Sistem menampilkan daftar data pasien</li> <li>5. Aktor menekan tombol Tambah Data</li> </ol>

	6. Sistem menampilkan <i>form</i> pendaftaran pasien 7. Aktor mengisi <i>form</i> pendaftaran pasien dan menekan tombol Simpan 8. Sistem berhasil menyimpan data pendaftaran pasien yang baru ke dalam <i>database</i> kemudian sistem menampilkan kembali daftar data pasien
Alternative flow	1. Data form ( <i>required</i> ) tidak diisi dengan lengkap 2. Sistem akan menampilkan pesan "Field tidak boleh kosong"
Post condition	Sistem menyimpan data pendaftaran pasien yang baru ke dalam <i>database</i>

**Tabel 4.7 Use Case Scenario Lihat Daftar Data Pasien**

Lihat Daftar Data Pasien	
Objective	Aktor dapat melihat daftar data pasien
Actors	Perekam Medis
Pre condition	Aktor <i>login</i> ke dalam sistem
Main flow	1. Sistem menampilkan halaman beranda 2. Aktor menekan tombol data pasien 3. Sistem menampilkan daftar data pasien
Alternative flow	-
Post condition	Sistem menampilkan daftar data pasien

**Tabel 4.8 Use Case Scenario Edit Data Pendaftaran Pasien**

Edit Data Pendaftaran Pasien	
Objective	Aktor dapat mengedit data pendaftaran pasien
Actors	Perekam Medis
Pre condition	Aktor melihat daftar data pasien
Main flow	1. Sistem menampilkan daftar data pasien 2. Aktor menekan tombol edit pada pasien yang datanya akan diedit 3. Sistem menampilkan <i>form</i> edit data pasien 4. Aktor mengisi <i>form</i> edit data pasien dan menekan tombol Simpan 5. Sistem berhasil mengupdate data pasien di <i>database</i> kemudian sistem menampilkan kembali daftar data pasien
Alternative flow	-
Post condition	Sistem menyimpan data pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali daftar data pasien



Tabel 4.9 Use Case Scenario Cari Data Pasien

Cari Data Pasien	
Objective	Aktor mencari data pasien
Actors	Perekam Medis
Pre condition	Aktor melihat daftar data pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan daftar data pasien</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan daftar data pasien sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan "<i>No matching records found</i>"</li> </ol>
Post condition	Sistem menampilkan kembali daftar data pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>

Tabel 4.10 Use Case Scenario Tambah Data Ringkasan Masuk dan Keluar

Tambah Data Ringkasan Masuk dan Keluar	
Objective	Aktor menambahkan data ringkasan masuk dan keluar dengan mengisi <i>form</i> ringkasan masuk dan keluar
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol Tambah Data</li> <li>2. Sistem menampilkan <i>form</i> ringkasan masuk dan keluar</li> <li>3. Aktor mengisi <i>form</i> ringkasan masuk dan keluar dan kemudian menekan tombol Simpan</li> <li>4. Sistem berhasil menyimpan data ringkasan masuk dan keluar ke dalam database dan menampilkan kembali daftar data ringkasan masuk dan keluar pasien</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Data <i>form</i> tidak diisi dengan lengkap</li> <li>2. Sistem akan menampilkan pesan "<i>Please fill out this field</i>"</li> </ol>
Post condition	Sistem menyimpan data ringkasan masuk dan keluar yang baru ke dalam <i>database</i>

**Tabel 4.11 Use Case Scenario Lihat Data Ringkasan Masuk dan Keluar**

<b>Lihat Data Ringkasan Masuk dan Keluar</b>	
Objective	Aktor dapat melihat daftar data ringkasan masuk dan keluar pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data pasien</li> <li>2. Aktor menekan tombol buka dan menekan tombol RMK</li> <li>3. Sistem menampilkan data ringkasan masuk dan keluar</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan data ringkasan masuk dan keluar pasien

**Tabel 4.12 Use Case Scenario Lihat Detail Data Ringkasan Masuk dan Keluar**

<b>Lihat Detail Data Ringkasan Masuk dan Keluar</b>	
Objective	Aktor dapat melihat detail data ringkasan masuk dan keluar pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data ringkasan masuk dan keluar pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data ringkasan masuk dan keluar pasien</li> <li>2. Aktor menekan tombol buka</li> <li>3. Sistem menampilkan detail data ringkasan masuk dan keluar</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan detail data ringkasan masuk dan keluar pasien

**Tabel 4.13 Use Case Scenario Edit Data Ringkasan Masuk dan Keluar Pasien**

<b>Edit Data Ringkasan Masuk dan Keluar Pasien</b>	
Objective	Aktor dapat mengedit data ringkasan masuk dan keluar pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data ringkasan masuk dan keluar pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data ringkasan masuk dan keluar pasien</li> <li>2. Aktor menekan tombol edit pada data ringkasan masuk dan keluar yang akan diedit</li> </ol>

	<ol style="list-style-type: none"> <li>3. Sistem menampilkan <i>form</i> edit data ringkasan masuk dan keluar pasien</li> <li>4. Aktor mengisi <i>form</i> edit data ringkasan masuk dan keluar pasien dan menekan tombol Simpan</li> <li>5. Sistem berhasil mengupdate data ringkasan masuk dan keluar pasien di <i>database</i> kemudian sistem menampilkan kembali data ringkasan masuk dan keluar pasien</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data ringkasan masuk dan keluar pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data ringkasan masuk dan keluar pasien

**Tabel 4.14 Use Case Scenario Cari Data Ringkasan Masuk dan Keluar Pasien**

<b>Cari Data Ringkasan Masuk dan Keluar Pasien</b>	
Objective	Aktor mencari data ringkasan masuk dan keluar pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data ringkasan masuk dan keluar pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data ringkasan masuk dan keluar pasien</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan data ringkasan masuk dan keluar pasien sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"</li> </ol>
Post condition	Sistem menampilkan kembali data ringkasan masuk dan keluar pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>

**Tabel 4.15 Use Case Scenario Tambah Data Resume Medis**

<b>Tambah Data Resume Medis</b>	
Objective	Aktor menambahkan data resume medis dengan mengisi <i>form</i> resume medis
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol Tambah Data</li> <li>2. Sistem menampilkan <i>form</i> resume medis</li> </ol>

	3. Aktor mengisi <i>form</i> resume medis dan kemudian menekan tombol Simpan 4. Sistem berhasil menyimpan data resume medis ke dalam database dan menampilkan kembali daftar data resume medis pasien
Alternative flow	1. Data <i>form</i> tidak diisi dengan lengkap 2. Sistem akan menampilkan pesan " <i>Please fill out this field</i> "
Post condition	Sistem menyimpan data resume medis yang baru ke dalam <i>database</i>

**Tabel 4.16 Use Case Scenario Lihat Data Resume Medis**

Lihat Data Resume Medis	
Objective	Aktor dapat melihat daftar data resume medis pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	1. Sistem menampilkan halaman daftar data pasien 2. Aktor menekan tombol buka dan menekan tombol RM 3. Sistem menampilkan data resume medis
Alternative flow	-
Post condition	Sistem menampilkan data resume medis pasien

**Tabel 4.17 Use Case Scenario Lihat Detail Data Resume Medis**

Lihat Detail Data Resume Medis	
Objective	Aktor dapat melihat detail data resume medis pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data resume medis pasien
Main flow	1. Sistem menampilkan halaman daftar data resume medis pasien 2. Aktor menekan tombol buka 3. Sistem menampilkan detail data resume medis
Alternative flow	-
Post condition	Sistem menampilkan detail data resume medis

**Tabel 4.18 Use Case Scenario Edit Data Resume Medis Pasien**

Edit Data Resume Medis Pasien	
Objective	Aktor dapat mengedit data resume medis pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data resume medis pasien
Main flow	1. Sistem menampilkan data resume medis pasien

	<ol style="list-style-type: none"> <li>Aktor menekan tombol edit pada data resume medis yang akan diedit</li> <li>Sistem menampilkan <i>form</i> edit data resume medis pasien</li> <li>Aktor mengisi <i>form</i> edit data resume medis pasien dan menekan tombol Simpan</li> <li>Sistem berhasil mengupdate data resume medis pasien di <i>database</i> kemudian sistem menampilkan kembali data resume medis pasien</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data resume medis pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data resume medis pasien

**Tabel 4.19 Use Case Scenario Cari Data Resume Medis Pasien**

Cari Data Resume Medis Pasien	
Objective	Aktor mencari data resume medis pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data resume medis pasien
Main flow	<ol style="list-style-type: none"> <li>Sistem menampilkan data resume medis pasien</li> <li>Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>Sistem menampilkan data resume medis pasien sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>Kata kunci yang dicari tidak ditemukan</li> <li>Sistem tidak menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"</li> </ol>
Post condition	Sistem menampilkan kembali data resume medis pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>

**Tabel 4.20 Use Case Scenario Tambah Data Rencana Pulang**

Tambah Data Rencana Pulang	
Objective	Aktor menambahkan data rencana pulang dengan mengisi <i>form</i> rencana pulang
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>Aktor menekan tombol Tambah Data</li> <li>Sistem menampilkan <i>form</i> rencana pulang</li> <li>Aktor mengisi <i>form</i> rencana pulang dan kemudian menekan tombol Simpan</li> </ol>

	4. Sistem berhasil menyimpan data rencana pulang ke dalam database dan menampilkan kembali daftar data rencana pulang pasien
Alternative flow	1. Data <i>form</i> tidak diisi dengan lengkap 2. Sistem akan menampilkan pesan " <i>Please fill out this field</i> "
Post condition	Sistem menyimpan data rencana pulang yang baru ke dalam <i>database</i>

**Tabel 4.21 Use Case Scenario Lihat Data Rencana Pulang**

Lihat Data Rencana Pulang	
Objective	Aktor dapat melihat daftar data rencana pulang pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	1. Sistem menampilkan halaman daftar data pasien 2. Aktor menekan tombol buka dan menekan tombol RP 3. Sistem menampilkan data rencana pulang
Alternative flow	-
Post condition	Sistem menampilkan data rencana pulang pasien

**Tabel 4.22 Use Case Scenario Lihat Detail Data Rencana Pulang**

Lihat Detail Data Rencana Pulang	
Objective	Aktor dapat melihat detail data rencana pulang pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rencana pulang pasien
Main flow	1. Sistem menampilkan halaman daftar data rencana pulang pasien 2. Aktor menekan tombol buka 3. Sistem menampilkan detail data rencana pulang
Alternative flow	-
Post condition	Sistem menampilkan detail data rencana pulang

**Tabel 4.23 Use Case Scenario Edit Data Rencana Pulang Pasien**

Edit Data Rencana Pulang Pasien	
Objective	Aktor dapat mengedit data rencana pulang pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rencana pulang pasien
Main flow	1. Sistem menampilkan data rencana pulang pasien 2. Aktor menekan tombol edit pada data rencana pulang yang akan diedit



	<ol style="list-style-type: none"> <li>3. Sistem menampilkan <i>form</i> edit data rencana pulang pasien</li> <li>4. Aktor mengisi <i>form</i> edit data rencana pulang pasien dan menekan tombol Simpan</li> <li>5. Sistem berhasil mengupdate data rencana pulang pasien di <i>database</i> kemudian sistem menampilkan kembali data rencana pulang pasien</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data rencana pulang pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data rencana pulang pasien

**Tabel 4.24 Use Case Scenario Cari Data Rencana Pulang Pasien**

<b>Cari Data Rencana Pulang Pasien</b>	
Objective	Aktor mencari data rencana pulang pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rencana pulang pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data rencana pulang pasien</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan data rencana pulang pasien sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"</li> </ol>
Post condition	Sistem menampilkan kembali data rencana pulang pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>

**Tabel 4.25 Use Case Scenario Tambah Data Pengkajian Asuhan Kebidanan**

<b>Tambah Data Pengkajian Asuhan Kebidanan</b>	
Objective	Aktor menambahkan data pengkajian asuhan kebidanan dengan mengisi <i>form</i> pengkajian asuhan kebidanan
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol Tambah Data</li> <li>2. Sistem menampilkan <i>form</i> pengkajian asuhan kebidanan</li> <li>3. Aktor mengisi <i>form</i> pengkajian asuhan kebidanan dan kemudian menekan tombol Simpan</li> </ol>

	4. Sistem berhasil menyimpan data pengkajian asuhan kebidanan ke dalam database dan menampilkan kembali daftar data pengkajian asuhan kebidanan pasien
Alternative flow	1. Data <i>form</i> tidak diisi dengan lengkap 2. Sistem akan menampilkan pesan " <i>Please fill out this field</i> "
Post condition	Sistem menyimpan data pengkajian asuhan kebidanan yang baru ke dalam <i>database</i>

Tabel 4.26 Use Case Scenario Lihat Data Pengkajian Asuhan Kebidanan

Lihat Data Pengkajian Asuhan Kebidanan	
Objective	Aktor dapat melihat daftar data pengkajian asuhan kebidanan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	1. Sistem menampilkan halaman daftar data pasien 2. Aktor menekan tombol buka dan menekan tombol AKBID 3. Sistem menampilkan data pengkajian asuhan kebidanan
Alternative flow	-
Post condition	Sistem menampilkan data pengkajian asuhan kebidanan pasien

Tabel 4.27 Use Case Scenario Lihat Detail Data Pengkajian Asuhan Kebidanan

Lihat Detail Data Pengkajian Asuhan Kebidanan	
Objective	Aktor dapat melihat detail data pengkajian asuhan kebidanan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data pengkajian asuhan kebidanan pasien
Main flow	1. Sistem menampilkan halaman daftar data pengkajian asuhan kebidanan pasien 2. Aktor menekan tombol buka 3. Sistem menampilkan detail data pengkajian asuhan kebidanan
Alternative flow	-
Post condition	Sistem menampilkan detail data pengkajian asuhan kebidanan

**Tabel 4.28 Use Case Scenario Edit Data Pengkajian Asuhan Kebidanan Pasien**

<b>Edit Data Pengkajian Asuhan Kebidanan Pasien</b>	
Objective	Aktor dapat mengedit data pengkajian asuhan kebidanan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data pengkajian asuhan kebidanan pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data pengkajian asuhan kebidanan pasien</li> <li>2. Aktor menekan tombol edit pada data pengkajian asuhan kebidanan yang akan diedit</li> <li>3. Sistem menampilkan <i>form</i> edit data pengkajian asuhan kebidanan pasien</li> <li>4. Aktor mengisi <i>form</i> edit data pengkajian asuhan kebidanan pasien dan menekan tombol Simpan</li> <li>5. Sistem berhasil mengupdate data pengkajian asuhan kebidanan pasien di <i>database</i> kemudian sistem menampilkan kembali data pengkajian asuhan kebidanan pasien</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data pengkajian asuhan kebidanan pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data pengkajian asuhan kebidanan pasien

**Tabel 4.29 Use Case Scenario Cari Data Pengkajian Asuhan Kebidanan Pasien**

<b>Cari Data Pengkajian Asuhan Kebidanan Pasien</b>	
Objective	Aktor mencari data pengkajian asuhan kebidanan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data pengkajian asuhan kebidanan pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data pengkajian asuhan kebidanan pasien</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan data pengkajian asuhan kebidanan pasien sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"</li> </ol>

Post condition	Sistem menampilkan kembali data pengkajian asuhan kebidanan pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>
----------------	---

**Tabel 4.30 Use Case Scenario Tambah Data Pengkajian Asuhan Keperawatan**

Tambah Data Pengkajian Asuhan Keperawatan	
Objective	Aktor menambahkan data pengkajian asuhan keperawatan dengan mengisi <i>form</i> pengkajian asuhan keperawatan
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol Tambah Data</li> <li>2. Sistem menampilkan <i>form</i> pengkajian asuhan keperawatan</li> <li>3. Aktor mengisi <i>form</i> pengkajian asuhan keperawatan dan kemudian menekan tombol Simpan</li> <li>4. Sistem berhasil menyimpan data pengkajian asuhan keperawatan ke dalam database dan menampilkan kembali daftar data pengkajian asuhan keperawatan pasien</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Data <i>form</i> tidak diisi dengan lengkap</li> <li>2. Sistem akan menampilkan pesan "<i>Please fill out this field</i>"</li> </ol>
Post condition	Sistem menyimpan data pengkajian asuhan keperawatan yang baru ke dalam <i>database</i>

**Tabel 4.31 Use Case Scenario Lihat Data Pengkajian Asuhan Keperawatan**

Lihat Data Pengkajian Asuhan Keperawatan	
Objective	Aktor dapat melihat daftar data pengkajian asuhan keperawatan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data pasien</li> <li>2. Aktor menekan tombol buka dan menekan tombol AKPER</li> <li>3. Sistem menampilkan data pengkajian asuhan keperawatan</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan data pengkajian asuhan keperawatan pasien

**Tabel 4.32 Use Case Scenario Lihat Detail Data Pengkajian Asuhan Keperawatan**

<b>Lihat Detail Data Pengkajian Asuhan Keperawatan</b>	
Objective	Aktor dapat melihat detail data pengkajian asuhan keperawatan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data pengkajian asuhan keperawatan pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data pengkajian asuhan keperawatan pasien</li> <li>2. Aktor menekan tombol buka</li> <li>3. Sistem menampilkan detail data pengkajian asuhan keperawatan</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan detail data pengkajian asuhan keperawatan

**Tabel 4.33 Use Case Scenario Edit Data Pengkajian Asuhan Keperawatan Pasien**

<b>Edit Data Pengkajian Asuhan Keperawatan Pasien</b>	
Objective	Aktor dapat mengedit data pengkajian asuhan keperawatan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data pengkajian asuhan keperawatan pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data pengkajian asuhan keperawatan pasien</li> <li>2. Aktor menekan tombol edit pada data pengkajian asuhan keperawatan yang akan diedit</li> <li>3. Sistem menampilkan <i>form</i> edit data pengkajian asuhan keperawatan pasien</li> <li>4. Aktor mengisi <i>form</i> edit data pengkajian asuhan keperawatan pasien dan menekan tombol Simpan</li> <li>5. Sistem berhasil mengupdate data pengkajian asuhan keperawatan pasien di <i>database</i> kemudian sistem menampilkan kembali data pengkajian asuhan keperawatan pasien</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data pengkajian asuhan keperawatan pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data pengkajian asuhan keperawatan pasien

**Tabel 4.34 Use Case Scenario Cari Data Pengkajian Asuhan Keperawatan Pasien**

<b>Cari Data Pengkajian Asuhan Keperawatan Pasien</b>	
Objective	Aktor mencari data pengkajian asuhan keperawatan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data pengkajian asuhan keperawatan pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data pengkajian asuhan keperawatan pasien</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan data pengkajian asuhan keperawatan pasien sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"</li> </ol>
Post condition	Sistem menampilkan kembali data pengkajian asuhan keperawatan pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>

**Tabel 4.35 Use Case Scenario Tambah Data MR3.1**

<b>Tambah Data MR3.1</b>	
Objective	Aktor menambahkan data MR3.1 dengan mengisi <i>form</i> MR3.1
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol Tambah Data</li> <li>2. Sistem menampilkan <i>form</i> MR3.1</li> <li>3. Aktor mengisi <i>form</i> MR3.1 dan kemudian menekan tombol Simpan</li> <li>4. Sistem berhasil menyimpan data MR3.1 ke dalam database dan menampilkan kembali daftar data MR3.1 pasien</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Data form (<i>required</i>) tidak diisi dengan lengkap</li> <li>2. Sistem akan menampilkan pesan "<i>Please fill out this field</i>"</li> </ol>
Post condition	Sistem menyimpan data MR3.1 yang baru ke dalam <i>database</i>



Tabel 4.36 Use Case Scenario Lihat Data MR3.1

Lihat Data Pengkajian MR3.1	
Objective	Aktor dapat melihat daftar data MR3.1 pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data pasien</li> <li>2. Aktor menekan tombol buka dan menekan tombol MR3.1</li> <li>4. Sistem menampilkan data MR3.1</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan data MR3.1 pasien

Tabel 4.37 Use Case Scenario Lihat Detail Data MR3.1

Lihat Detail Data MR3.1	
Objective	Aktor dapat melihat detail data MR3.1 pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data MR3.1 pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data MR3.1 pasien</li> <li>2. Aktor menekan tombol buka</li> <li>3. Sistem menampilkan detail data MR3.1</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan detail data MR3.1

Tabel 4.38 Use Case Scenario Edit Data MR3.1 Pasien

Edit Data MR3.1 Pasien	
Objective	Aktor dapat mengedit data MR3.1 pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data MR3.1 pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data MR3.1 pasien</li> <li>2. Aktor menekan tombol edit pada data MR3.1 yang akan diedit</li> <li>3. Sistem menampilkan <i>form</i> edit data MR3.1 pasien</li> <li>4. Aktor mengisi <i>form</i> edit data MR3.1 pasien dan menekan tombol Simpan</li> <li>5. Sistem berhasil mengupdate data MR3.1 pasien di <i>database</i> kemudian sistem menampilkan kembali data MR3.1 pasien</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data MR3.1 pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data MR3.1 pasien

Tabel 4.39 Use Case Scenario Cari Data MR3.1 Pasien

Cari Data MR3.1 Pasien	
Objective	Aktor mencari data MR3.1 pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data MR3.1 pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data MR3.1 pasien</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan data MR3.1 pasien sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"</li> </ol>
Post condition	Sistem menampilkan kembali data MR3.1 pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>

Tabel 4.40 Use Case Scenario Tambah Data MR3.2

Tambah Data MR3.2	
Objective	Aktor menambahkan data MR3.2 dengan mengisi <i>form</i> MR3.2
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol Tambah Data</li> <li>2. Sistem menampilkan <i>form</i> MR3.2</li> <li>3. Aktor mengisi <i>form</i> MR3.2 dan kemudian menekan tombol Simpan</li> <li>5. Sistem berhasil menyimpan data MR3.2 ke dalam database dan menampilkan kembali daftar data MR3.2 pasien</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Data <i>form</i> tidak diisi dengan lengkap</li> <li>2. Sistem akan menampilkan pesan "<i>Please fill out this field</i>"</li> </ol>
Post condition	Sistem menyimpan data MR3.2 yang baru ke dalam <i>database</i>

Tabel 4.41 Use Case Scenario Lihat Data MR3.2

Lihat Data Pengkajian MR3.2	
Objective	Aktor dapat melihat daftar data MR3.2 pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data pasien</li> <li>2. Aktor menekan tombol buka dan menekan tombol MR3.2</li> </ol>

	3. Sistem menampilkan data MR3.2
Alternative flow	-
Post condition	Sistem menampilkan data MR3.2 pasien

**Tabel 4.42 Use Case Scenario Lihat Detail Data MR3.2**

<b>Lihat Detail Data MR3.2</b>	
Objective	Aktor dapat melihat detail data MR3.2 pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data MR3.2 pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data MR3.2 pasien</li> <li>2. Aktor menekan tombol buka</li> <li>3. Sistem menampilkan detail data MR3.2</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan detail data MR3.2

**Tabel 4.43 Use Case Scenario Edit Data MR3.2 Pasien**

<b>Edit Data MR3.2 Pasien</b>	
Objective	Aktor dapat mengedit data MR3.2 pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data MR3.2 pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data MR3.2 pasien</li> <li>2. Aktor menekan tombol edit pada data MR3.2 yang akan diedit</li> <li>3. Sistem menampilkan <i>form</i> edit data MR3.2 pasien</li> <li>4. Aktor mengisi <i>form</i> edit data MR3.2 pasien dan menekan tombol Simpan</li> <li>5. Sistem berhasil mengupdate data MR3.2 pasien di <i>database</i> kemudian sistem menampilkan kembali data MR3.2 pasien</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data MR3.2 pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data MR3.2 pasien

**Tabel 4.44 Use Case Scenario Cari Data MR3.2 Pasien**

<b>Cari Data MR3.2 Pasien</b>	
Objective	Aktor mencari data MR3.2 pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data MR3.2 pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data MR3.2 pasien</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> </ol>

	3. Sistem menampilkan data MR3.2 pasien sesuai dengan kata kunci yang dimasukkan
Alternative flow	1. Kata kunci yang dicari tidak ditemukan 2. Sistem menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"
Post condition	Sistem menampilkan kembali data MR3.2 pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>

**Tabel 4.45 Use Case Scenario Tambah Data Rawat Jalan**

Tambah Data Rawat Jalan	
Objective	Aktor menambahkan data rawat jalan dengan mengisi <i>form</i> rawat jalan
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	1. Aktor menekan tombol Tambah Data 2. Sistem menampilkan <i>form</i> rawat jalan 3. Aktor mengisi <i>form</i> rawat jalan dan kemudian menekan tombol Simpan 4. Sistem berhasil menyimpan data rawat jalan ke dalam database dan menampilkan kembali daftar data rawat jalan pasien
Alternative flow	1. Data <i>form</i> tidak diisi dengan lengkap 2. Sistem akan menampilkan pesan " <i>Please fill out this field</i> "
Post condition	Sistem menyimpan data rawat jalan yang baru ke dalam <i>database</i>

**Tabel 4.46 Use Case Scenario Lihat Data Rawat Jalan**

Lihat Data Rawat Jalan	
Objective	Aktor dapat melihat daftar data rawat jalan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	1. Sistem menampilkan halaman daftar data pasien 2. Aktor menekan tombol buka dan menekan tombol rawat jalan 3. Sistem menampilkan data rawat jalan
Alternative flow	-
Post condition	Sistem menampilkan data rawat jalan pasien

**Tabel 4.47 Use Case Scenario Lihat Detail Data Rawat Jalan**

Lihat Detail Data Rawat Jalan	
Objective	Aktor dapat melihat detail data rawat jalan pasien

Actors	Perekam Medis
Pre condition	Aktor melihat data rawat jalan pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data rawat jalan pasien</li> <li>2. Aktor menekan tombol buka</li> <li>3. Sistem menampilkan detail data rawat jalan</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan detail data rawat jalan

Tabel 4.48 Use Case Scenario Edit Data Rawat Jalan Pasien

Edit Data Rawat Jalan Pasien	
Objective	Aktor dapat mengedit data rawat jalan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rawat jalan pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data rawat jalan pasien</li> <li>2. Aktor menekan tombol edit pada data rawat jalan yang akan diedit</li> <li>3. Sistem menampilkan <i>form</i> edit data rawat jalan pasien</li> <li>4. Aktor mengisi <i>form</i> edit data rawat jalan pasien dan menekan tombol Simpan</li> <li>5. Sistem berhasil mengupdate data rawat jalan pasien di <i>database</i> kemudian sistem menampilkan kembali data rawat jalan pasien</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data rawat jalan pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data rawat jalan pasien

Tabel 4.49 Use Case Scenario Cari Data Rawat Jalan Pasien

Cari Data Rawat Jalan Pasien	
Objective	Aktor mencari data rawat jalan pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rawat jalan pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data rawat jalan pasien</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan data rawat jalan pasien sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"</li> </ol>

Post condition	Sistem menampilkan kembali data rawat jalan pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>
----------------	---

**Tabel 4.50 Use Case Scenario Tambah Data Sebab Kematian**

Tambah Data Sebab Kematian	
Objective	Aktor menambahkan data sebab kematian dengan mengisi <i>form</i> sebab kematian
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol Tambah Data</li> <li>2. Sistem menampilkan <i>form</i> sebab kematian</li> <li>3. Aktor mengisi <i>form</i> sebab kematian dan kemudian menekan tombol Simpan</li> <li>4. Sistem berhasil menyimpan data sebab kematian ke dalam database dan menampilkan kembali daftar data sebab kematian pasien</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Data <i>form</i> tidak diisi dengan lengkap</li> <li>2. Sistem akan menampilkan pesan "<i>Please fill out this field</i>"</li> </ol>
Post condition	Sistem menyimpan data sebab kematian ke dalam <i>database</i>

**Tabel 4.51 Use Case Scenario Lihat Data Sebab Kematian**

Lihat Data Sebab Kematian	
Objective	Aktor dapat melihat data sebab kematian pasien
Actors	Perekam Medis
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data pasien</li> <li>2. Aktor menekan tombol buka dan menekan tombol sebab kematian</li> <li>3. Sistem menampilkan data sebab kematian</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan data sebab kematian pasien

**Tabel 4.52 Use Case Scenario Cetak Laporan Persalinan**

Cetak Laporan Persalinan	
Objective	Aktor mencetak laporan persalinan
Actors	Perekam Medis
Pre condition	Aktor mengakses halaman laporan persalinan
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan <i>form</i> laporan persalinan</li> </ol>



	<ol style="list-style-type: none"> <li>Aktor mengisi unit pelayanan, periode laporan, dan nama pelapor kemudian menekan tombol Cetak</li> <li>Sistem menampilkan <i>preview</i> halaman laporan persalinan dalam bentuk pdf</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>Data <i>form</i> tidak diisi dengan lengkap</li> <li>Sistem akan menampilkan pesan "<i>Please fill out this field</i>"</li> </ol>
Post condition	Sistem menampilkan halaman laporan persalinan dalam bentuk pdf

**Tabel 4.53 Use Case Scenario Cetak Laporan Kematian Bayi**

<b>Cetak Laporan Kematian Bayi</b>	
Objective	Aktor mencetak laporan kematian bayi
Actors	Perekam Medis
Pre condition	Aktor mengakses halaman laporan kematian
Main flow	<ol style="list-style-type: none"> <li>Sistem menampilkan <i>form</i> laporan kematian bayi</li> <li>Aktor mengisi unit pelayanan, periode laporan, dan nama pelapor kemudian menekan tombol Cetak</li> <li>Sistem menampilkan <i>preview</i> halaman laporan kematian bayi dalam bentuk pdf</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>Data <i>form</i> tidak diisi dengan lengkap</li> <li>Sistem akan menampilkan pesan "<i>Please fill out this field</i>"</li> </ol>
Post condition	Sistem menampilkan halaman laporan kematian bayi dalam bentuk pdf

**Tabel 4.54 Use Case Scenario Tambah Data Visite**

<b>Tambah Data Visite</b>	
Objective	Aktor menambahkan data rawat jalan dengan mengisi <i>form</i> visite
Actors	Perawat
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>Aktor menekan tombol Tambah Data</li> <li>Sistem menampilkan <i>form</i> visite</li> <li>Aktor mengisi <i>form</i> visite dan kemudian menekan tombol Simpan</li> <li>Sistem berhasil menyimpan data visite ke dalam database dan menampilkan kembali daftar data visite pasien</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>Data <i>form</i> tidak diisi dengan lengkap</li> </ol>

	2. Sistem akan menampilkan pesan “ <i>Please fill out this field</i> ”
Post condition	Sistem menyimpan data rawat jalan yang baru ke dalam <i>database</i>

**Tabel 4.55 Use Case Scenario Lihat Data Visite**

Lihat Data Visite	
Objective	Aktor dapat melihat daftar data visite
Actors	Perawat
Pre condition	Aktor melihat data rekam medis pasien
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data pasien</li> <li>2. Aktor menekan tombol buka dan menekan tombol visite</li> <li>3. Sistem menampilkan data visite</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan data visite

**Tabel 4.56 Use Case Scenario Lihat Detail Data Visite**

Lihat Detail Data Visite	
Objective	Aktor dapat melihat detail data visite
Actors	Perawat
Pre condition	Aktor melihat data visite
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman daftar data visite</li> <li>2. Aktor menekan tombol buka</li> <li>3. Sistem menampilkan detail data visite</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan detail data visite

**Tabel 4.57 Use Case Scenario Edit Data Visite**

Edit Data Visite	
Objective	Aktor dapat mengedit data visite
Actors	Perawat
Pre condition	Aktor melihat data visite
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data visite</li> <li>2. Aktor menekan tombol edit pada data visite yang akan diedit</li> <li>3. Sistem menampilkan <i>form</i> edit data visite</li> <li>4. Aktor mengisi <i>form</i> edit data visite dan menekan tombol Simpan</li> <li>5. Sistem berhasil mengupdate datavisite di <i>database</i> kemudian sistem menampilkan kembali data visite</li> </ol>

Alternative flow	-
Post condition	Sistem menyimpan data visite pasien yang telah di edit ke dalam <i>database</i> dan menampilkan kembali data visite

**Tabel 4.58 Use Case Scenario Cari Data Visite**

Cari Data Visite	
Objective	Aktor mencari data visite
Actors	Perawat
Pre condition	Aktor melihat data visite
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan data visite</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan data visite sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan <i>notifications</i> "Data yang dicari tidak ditemukan"</li> </ol>
Post condition	Sistem menampilkan kembali data visite sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>

**Tabel 4.59 Use Case Scenario Tambah Data Dokter**

Tambah Data Dokter	
Objective	Aktor menambahkan data dokter dengan mengisi <i>form</i> tambah data dokter
Actors	Perekam Medis
Pre condition	Aktor melihat daftar data dokter
Main flow	<ol style="list-style-type: none"> <li>1. Aktor masuk ke halaman beranda</li> <li>2. Sistem menampilkan halaman beranda</li> <li>3. Aktor menekan tombol data dokter</li> <li>4. Sistem menampilkan daftar data dokter</li> <li>5. Aktor menekan tombol Tambah Data</li> <li>6. Sistem menampilkan <i>form</i> tambah data dokter</li> <li>7. Aktor mengisi <i>form</i> tambah data dokter dan menekan tombol Simpan</li> <li>8. Sistem berhasil menyimpan data dokter yang baru ke dalam <i>database</i> kemudian sistem menampilkan kembali daftar data dokter</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Data form (<i>required</i>) tidak diisi dengan lengkap</li> <li>2. Sistem akan menampilkan pesan "Field tidak boleh kosong"</li> </ol>
Post condition	Sistem menyimpan data dokter yang baru ke dalam <i>database</i>

**Tabel 4.60 Use Case Scenario Lihat Daftar Data Dokter**

<b>Lihat Daftar Data Dokter</b>	
Objective	Aktor dapat melihat daftar data dokter
Actors	Perekam Medis
Pre condition	Aktor <i>login</i> ke dalam sistem
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan halaman beranda</li> <li>2. Aktor menekan tombol data dokter</li> <li>3. Sistem menampilkan daftar data dokter</li> </ol>
Alternative flow	-
Post condition	Sistem menampilkan daftar data dokter

**Tabel 4.61 Use Case Scenario Edit Data Dokter**

<b>Edit Data Dokter</b>	
Objective	Aktor dapat mengedit data dokter
Actors	Perekam Medis
Pre condition	Aktor melihat daftar data dokter
Main flow	<ol style="list-style-type: none"> <li>6. Sistem menampilkan daftar data dokter</li> <li>7. Aktor menekan tombol edit pada data dokter yang datanya akan diedit</li> <li>8. Sistem menampilkan <i>form</i> edit data dokter</li> <li>9. Aktor mengisi <i>form</i> edit data dokter dan menekan tombol Simpan</li> <li>10. Sistem berhasil mengupdate data pasien di <i>database</i> kemudian sistem menampilkan kembali daftar data dokter</li> </ol>
Alternative flow	-
Post condition	Sistem menyimpan data dokter yang telah di edit ke dalam <i>database</i> dan menampilkan kembali daftar data dokter

**Tabel 4.62 Use Case Scenario Cari Data Dokter**

<b>Cari Data Dokter</b>	
Objective	Aktor mencari data dokter
Actors	Perekam Medis
Pre condition	Aktor melihat daftar data dokter
Main flow	<ol style="list-style-type: none"> <li>1. Sistem menampilkan daftar data dokter</li> <li>2. Aktor memasukkan kata kunci pada kolom pencarian dan menekan tombol Search</li> <li>3. Sistem menampilkan daftar data dokter sesuai dengan kata kunci yang dimasukkan</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>1. Kata kunci yang dicari tidak ditemukan</li> <li>2. Sistem menampilkan "<i>No matching records found</i>"</li> </ol>

Post condition	Sistem menampilkan kembali daftar data pasien sesuai dengan kata kunci yang dimasukkan oleh <i>actor</i>
----------------	--



## BAB 5 PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab ini akan dibahas mengenai perancangan dan implementasi sistem pada RSIA. Prof. H. M. Farid Makassar. Perancangan sistem terbagi dalam perancangan arsitektur, perancangan komponen, perancangan data, dan perancangan antarmuka. Dan pada implementasi sistem akan

### 5.1 Perancangan Sistem

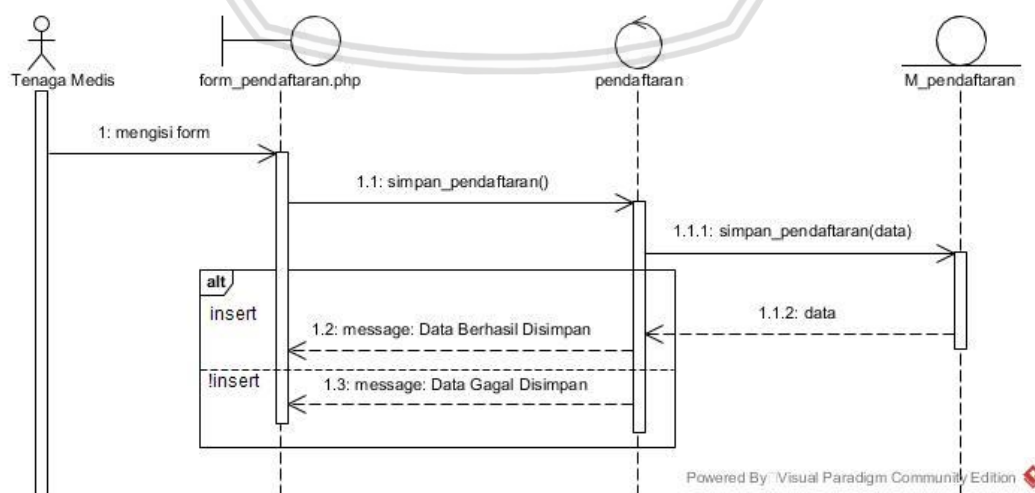
Perancangan sistem merupakan tahap untuk mendapatkan gambaran yang jelas mengenai apa yang akan dikerjakan berdasarkan analisa sistem. Perancangan sistem digunakan untuk memenuhi kebutuhan pemakaian sistem dan untuk memberikan gambaran yang jelas serta menghasilkan rancangan bangun yang lengkap.

#### 5.1.1 Perancangan Arsitektur

Perancangan arsitektur perangkat lunak merupakan proses desain untuk mengidentifikasi sub-sistem yang membentuk sebuah sistem. Perancangan arsitektur membangun struktur program menggunakan *class diagram* dan *sequence diagram*.

##### 5.1.1.1 Sequence Diagram

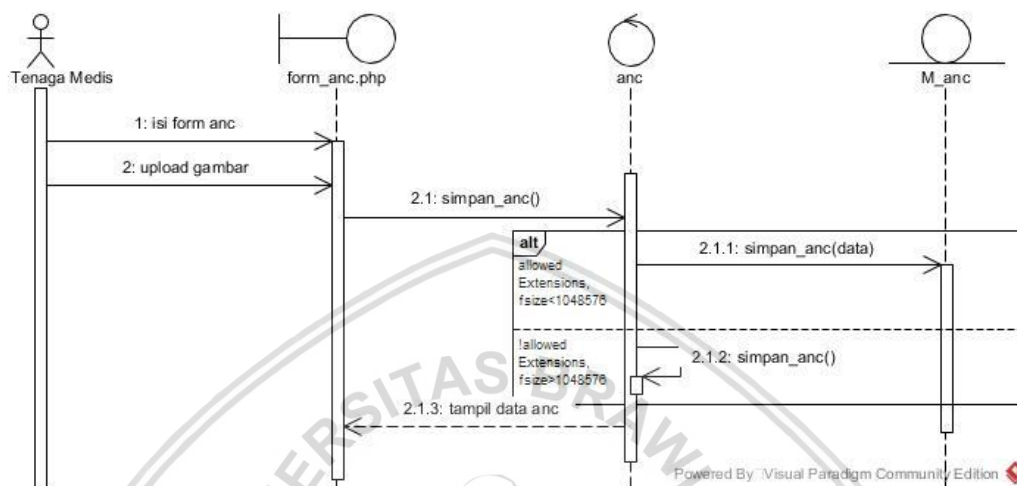
Gambar 5.1 merupakan gambar sequence diagram yang menunjukkan interaksi antara *actor* (perekam medis) pada proses tambah data pendaftaran pasien. Pada proses ini memiliki objek *boundary* yaitu list\_pendaftaran dan form\_pendaftaran, memiliki objek *controller* yaitu pendaftaran serta memiliki objek *entity* M\_pendaftaran. Untuk menjalankan alur *use case* tambah data pendaftaran pasien, aktor yang berperan sebagai *user* harus memilih fungsi tambah data pada halaman list\_pendaftaran.



Gambar 5.1 Sequence Diagram Tambah Data Pendaftaran Pasien

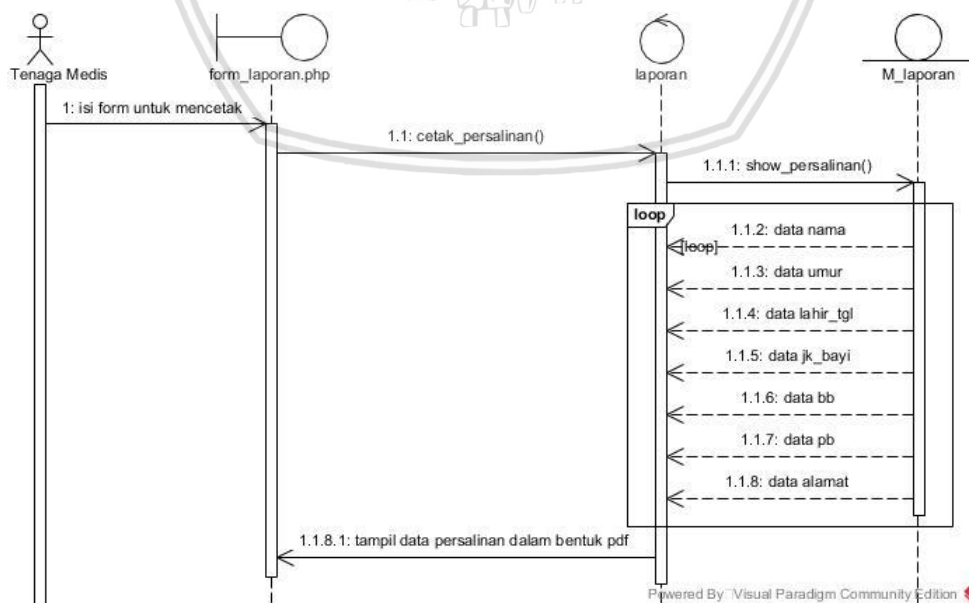


Gambar 5.2 merupakan gambar sequence diagram yang menunjukkan interaksi antara *actor* (perekam medis) pada proses tambah data rawat jalan khususnya untuk menyimpan file gambar. Pada proses ini memiliki objek *boundary* yaitu form\_anc, memiliki objek *controller* yaitu anc serta memiliki objek *entity* M\_anc. Untuk menjalankan alur *use case* tambah data rawat jalan, aktor yang berperan sebagai Perekam Medis harus membuka halaman form\_anc.



**Gambar 5.2 Sequence Diagram Tambah Data Rawat Jalan**

Gambar 5.3 merupakan gambar sequence diagram yang menunjukkan interaksi antara *actor* (perekam medis) pada proses cetak laporan persalinan. Pada proses ini memiliki objek *boundary* yaitu form\_laporan, memiliki objek *controller* yaitu laporan serta memiliki objek *entity* M\_laporan. Untuk menjalankan alur *use case* cetak laporan persalinan, aktor yang berperan sebagai *user* harus memilih fungsi cetak pada halaman form\_laporan.

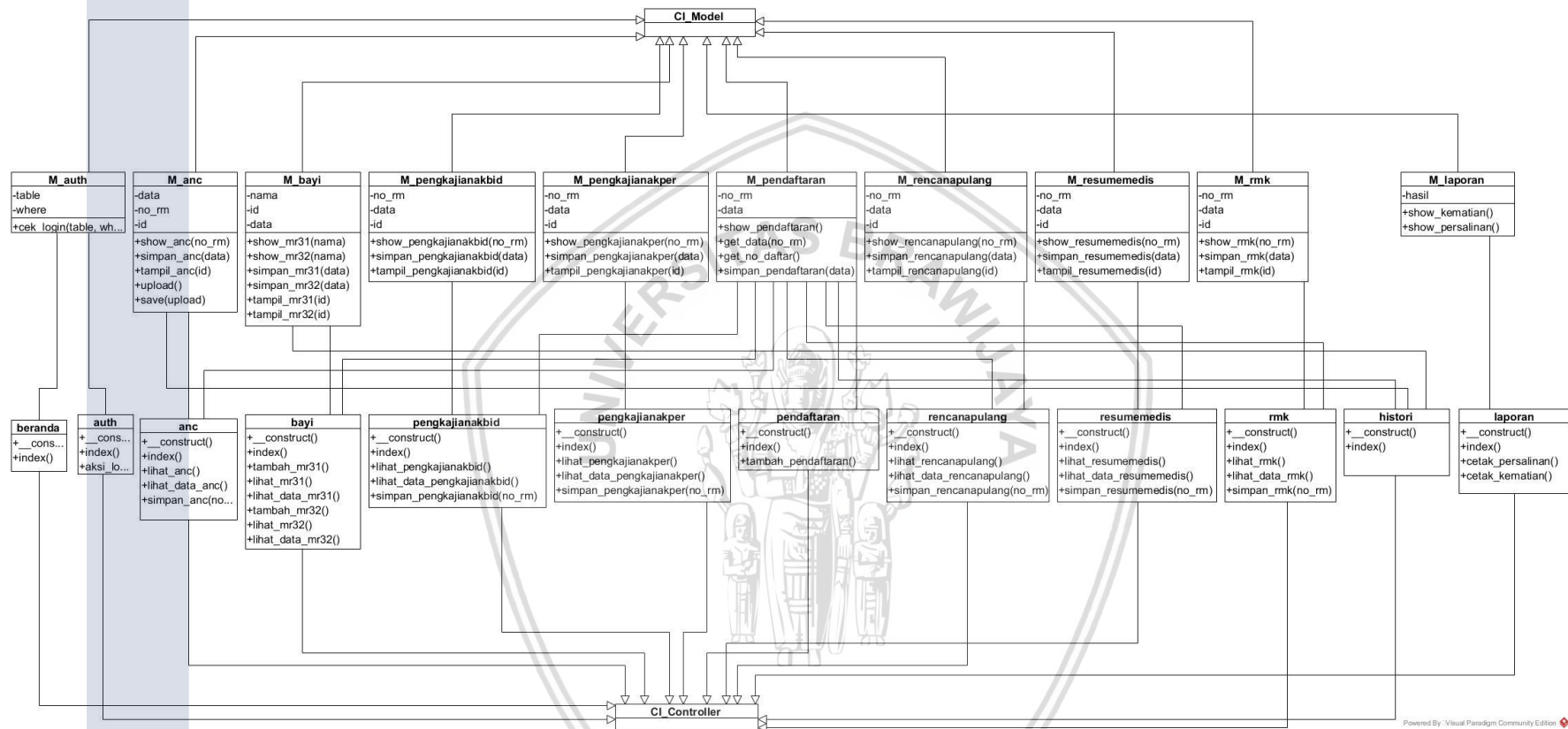


**Gambar 5.3 Sequence Diagram Cetak Laporan Persalinan**

#### 5.1.1.2 Class Diagram

Class diagram merupakan diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam sistem perangkat lunak yang akan dibangun. Class diagram menunjukkan hubungan antar kelas dalam sistem yang sedang dibangun dan bagaimana kelas-kelas tersebut saling berkolaborasi untuk mencapai sebuah tujuan. Gambar 5.4 pada halaman berikut merupakan gambaran class diagram dari aplikasi rekam medis RSIA. Prof. Dr. H. M Farid Makassar.





Gambar 5.4 Class Diagram Aplikasi Rekam Medis



**Tabel 5.1 Algoritma Method *simpan\_pendaftaran***

1	mulai method <i>simpan_pendaftaran</i>
2	if <i>form_pendaftaran</i> = false
3	masukkan data berdasarkan <i>no_daftar</i>
4	tampil view <i>form_pendaftaran</i>
5	else
6	masukkan data
7	memanggil method <i>simpan_pendaftaran</i> dari kelas <i>M_pendaftaran</i>
8	if (\$insert)
9	cetak data berhasil disimpan
10	else
11	cetak data gagal disimpan
12	end if
13	redirect('pendaftaran')
14	end if
15	end

Tabel 5.1 merupakan *pseudocode* *simpan pendaftaran*. Pada *pseudocode* ini diawali dengan pengecekan kondisi, apabila *form\_pendaftaran* bernilai false, maka aktor memasukkan data dengan *no\_daftar* yang diambil secara acak pada *form pendaftaran*. Kemudian memanggil *method* *simpan\_pendaftaran* dari kelas *M\_pendaftaran* dan dilakukan lagi pengecekan kondisi, kemudian kembali ke halaman data pendaftaran.

#### 5.1.3.2 Algoritma Method *simpan\_anc*

Nama Class : *anc.php*

Nama Operasi : *simpan\_anc()*

Algoritma :

**Tabel 5.2 Algoritma Method *simpan\_anc***

1	mulai method <i>simpan_anc</i>
2	pilih dan ambil data gambar
3	ambil value <i>fileInput</i> melalui <i>id</i>
4	ambil value <i>filePath</i> dari value <i>fileInput</i>
5	atur ekstensi file
6	if !allowedExtensions.exec( <i>filePath</i> )
7	tampil "File yang diupload hanya menggunakan ekstensi jpg, jpeg, png, dan gif"
8	set value <i>fileInput</i>
9	tampil tinjauan gambar di html
10	return false
11	else
12	ambil <i>fsize</i> gambar
13	if ( <i>fsize</i> >1048576)
14	tampil "File yang diupload hanya menggunakan ekstensi jpg, jpeg, png, dan gif"
	set value <i>fileInput</i>
	tampil tinjauan gambar di html

15	return false
16	else
17	if (fileInput.files &&
18	fileInput.files[0])
19	tampil tinjauan gambar di html
	end if
20	end if
21	end if
22	end
23	
24	

Tabel 5.2 merupakan *pseudocode* simpan anc dengan melakukan *upload* foto atau hasil USG. Pada *pseudocode* ini diawali dengan melakukan upload gambar kemudian dilakukan pengecekan kondisi untuk memastikan apakah gambar yang diupload mempunyai ekstensi yang sesuai. Kemudian dilakukan lagi pengecekan kondisi untuk memastikan ukuran atau *size* gambar tidak lebih dari 1MB lalu kemudian gambar yang dipilih ditampilkan.

### 5.1.3.3 Algoritma *Method* cetak\_persalinan

Nama Class : laporan.php

Nama Operasi : cetak\_persalinan()

Algoritma :

**Tabel 5.3 Algoritma *Method* cetak\_persalinan**

1	mulai method cetak_persalinan
2	memanggil method show_persalinan dari kelas M_laporan untuk mengambil data diantara tgl1 dan tgl2
3	no = 1
4	foreach row
5	no = ambil data no
6	nama = ambil data nama
7	umur = ambil data umur
8	lahir_tgl = ambil data tanggal lahir
9	jk_bayi = ambil data jenis kelamin bayi
10	bb = ambil data berat badan bayi
11	pb = ambil data panjang badan bayi
12	alamat = ambil data alamat
13	increment no
14	end foreach
15	end

Tabel 5.3 merupakan *pseudocode* cetak persalinan. Pada *pseudocode* ini diawali dengan pengambilan data dari *method* show\_persalinan pada kelas M\_laporan, kemudian dilakukan perulangan untuk mengambil data dari beberapa kolom.

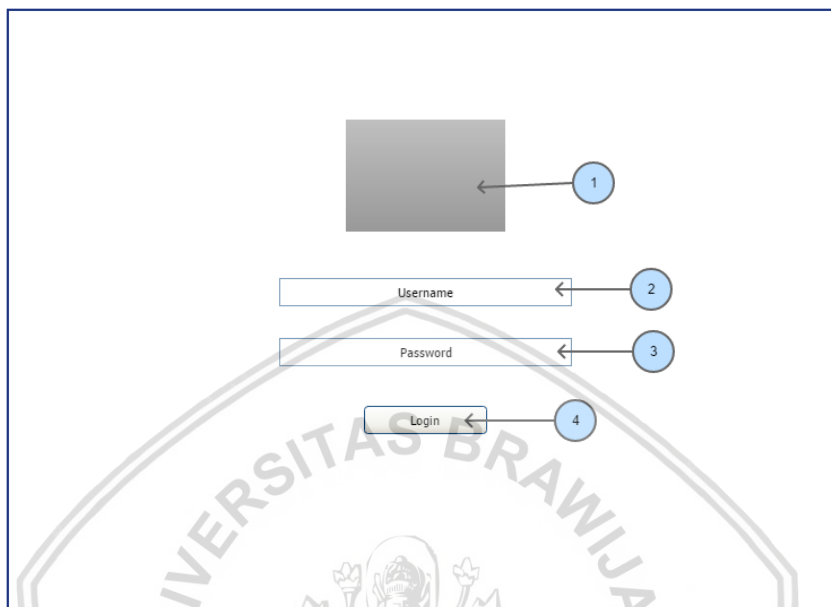
### 5.1.4 Perancangan Antarmuka

Perancangan antarmuka digunakan sebagai antarmuka antara modul perangkat lunak, antarmuka antara perangkat lunak dengan sumber informasi



selain manusia (*external entities*), serta antarmuka antara user dan perangkat lunak yang akan dibangun. Berikut ini merupakan perancangan antarmuka Aplikasi Rekam Medis Prof. Dr. H. M. Farid Makassar:

#### 5.1.4.1 Halaman Login



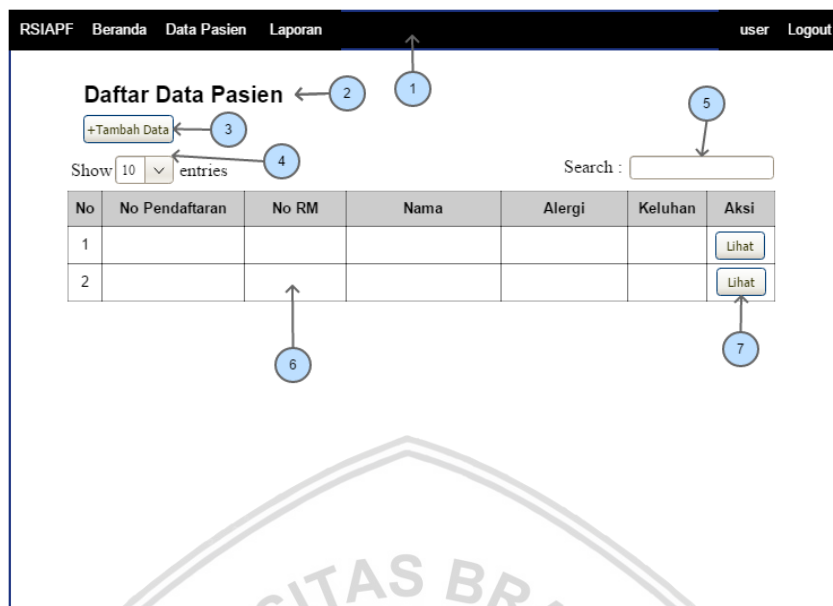
**Gambar 5.6 Antarmuka Halaman Login**

Gambar 5.6 adalah gambaran perancangan antarmuka halaman login. Untuk penjelasan dari objek antarmuka halaman login, dapat dilihat pada Tabel 5.6

**Tabel 5.4 Keterangan Antarmuka Halaman Login**

No.	Nama Objek	Tipe	Keterangan
1	logo	Image	Image untuk menampilkan logo rumah sakit
2	username	Input	Field yang digunakan untuk input username
3	password	Input	Field yang digunakan untuk input password
4	login	Button	Button yang digunakan untuk melakukan login

#### 5.1.4.2 Halaman Daftar Data Pasien



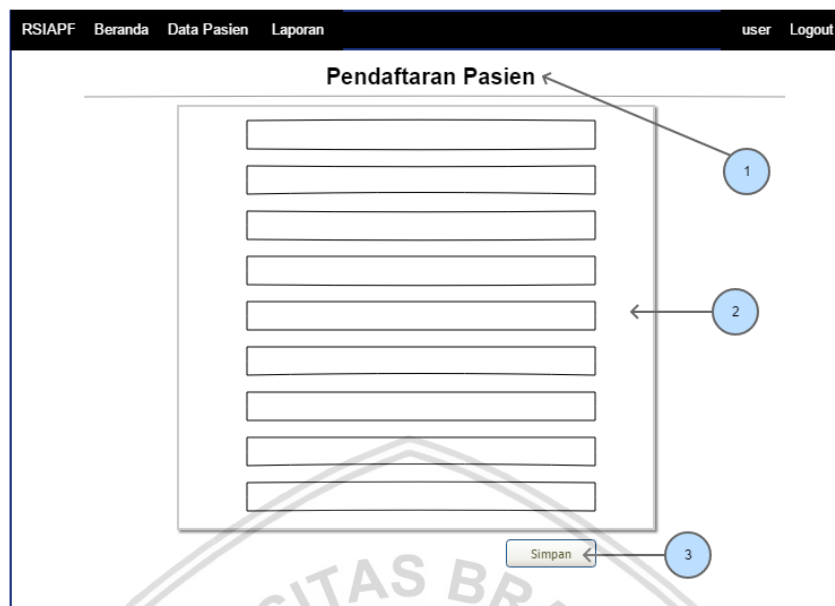
**Gambar 5.7 Antarmuka Halaman Daftar Data Pasien**

Gambar 5.7 adalah gambaran perancangan antarmuka halaman daftar data pasien. Untuk penjelasan dari objek antarmuka halaman daftar data pasien dapat dilihat pada Tabel 5.7

**Tabel 5.5 Keterangan Antarmuka Halaman Daftar Data Pasien**

No.	Nama Objek	Tipe	Keterangan
1	navbar	Nav	Navigation Bar untuk menampilkan menu
2	pageHeading	H2	Heading untuk menampilkan Daftar Data Pasien
3	tambah	Button	Tombol untuk melakukan tambah data pendaftaran pasien
4	comboSearch	Combobox	Combo box untuk memilih jumlah data yang ingin ditampilkan dalam satu halaman
5	search	Input	Text box untuk memasukkan keyword data yang akan dicari
6	datatable	Table	Tabel yang menampilkan data-data pasien
7	buka	Button	Tombol untuk melihat data-data rekam medis pasien

### 5.1.4.3 Halaman Pendaftaran Pasien



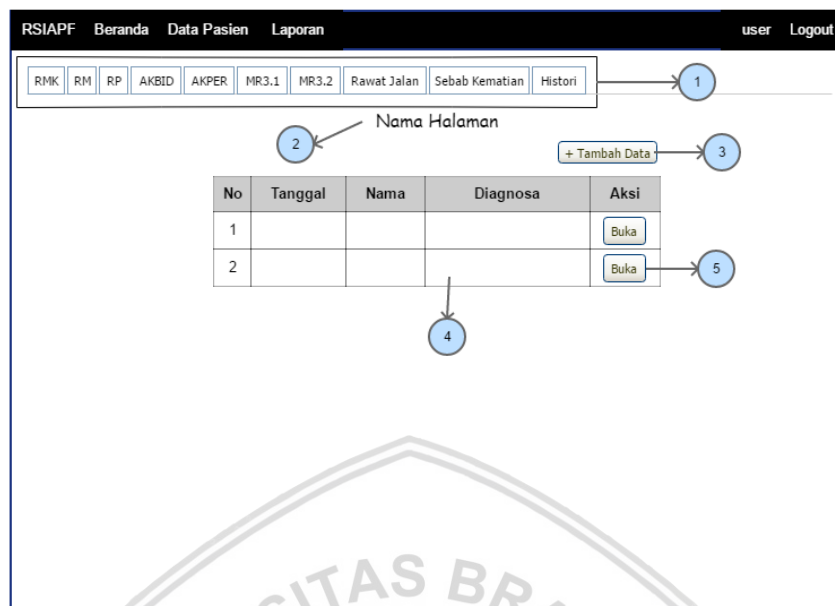
**Gambar 5.8 Antarmuka Halaman Pendaftaran Pasien**

Gambar 5.8 adalah gambaran perancangan antarmuka halaman pendaftaran pasien. Untuk penjelasan dari objek antarmuka halaman pendaftaran pasien dapat dilihat pada Tabel 5.8

**Tabel 5.6 Keterangan Antarmuka Halaman Pendaftaran Pasien**

No.	Nama Objek	Tipe	Keterangan
1	titlepageHeading	H2	Heading untuk menampilkan nama halaman Pendaftaran Pasien
2	form	Form	Form yang digunakan untuk memasukkan data pasien
3	simpan	Button	Button yang digunakan untuk menyimpan data pasien

#### 5.1.4.4 Halaman Data Rekam Medis



**Gambar 5.9 Antarmuka Halaman Data Rekam Medis**

Gambar 5.9 adalah gambaran perancangan antarmuka halaman data rekam medis. Untuk penjelasan dari objek antarmuka halaman data rekam medis dapat dilihat pada Tabel 5.9

**Tabel 5.7 Keterangan Antarmuka Halaman Data Rekam Medis**

No.	Nama Objek	Tipe	Keterangan
1	nav	Navtabs	Nav untuk menampilkan menu halaman rekam medis pasien
2	pagetitle	Text	Text yang digunakan untuk menampilkan judul halaman
3	tambah	Button	Button yang digunakan untuk melakukan tambah data
4	tabelrm	Table	Tabel yang digunakan untuk menampilkan daftar data rekam medis
5	buka	Button	Button yang digunakan untuk melihat detail data rekam medis

#### 5.1.4.5 Halaman Tambah Data Rekam Medis

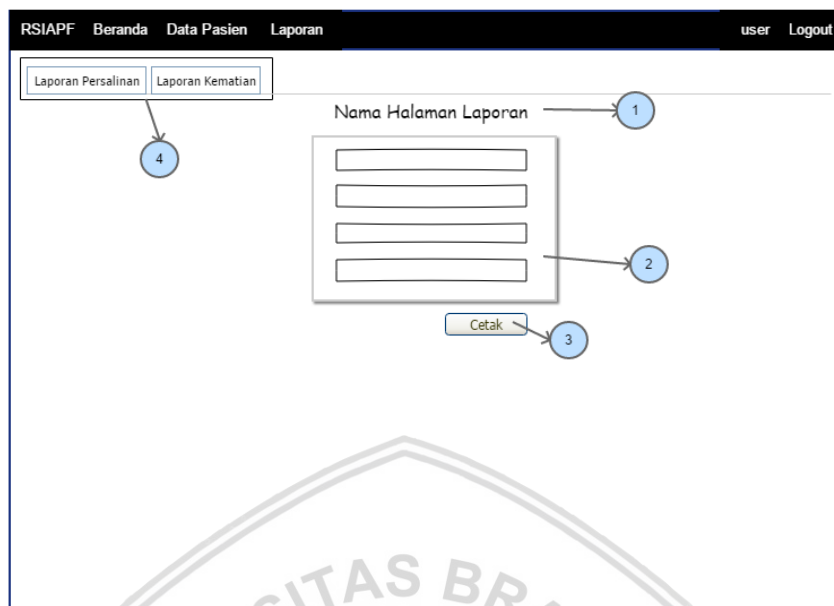
**Gambar 5.10 Antarmuka Halaman Tambah Data Rekam Medis**

Gambar 5.10 adalah gambaran perancangan antarmuka halaman tambah data rekam medis. Untuk penjelasan dari objek antarmuka halaman tambah data rekam medis dapat dilihat pada Tabel 5.10

**Tabel 5.8 Keterangan Antarmuka Halaman Tambah Data Rekam Medis**

No.	Nama Objek	Tipe	Keterangan
1	pagetitle	Text	Text untuk menampilkan judul halaman
2	formrm	Form	Form yang digunakan untuk memasukkan data rm
3	Simpan	Button	Button yang digunakan untuk menyimpan data

#### 5.1.4.6 Halaman Laporan



**Gambar 5.11 Antarmuka Halaman Laporan**

Gambar 5.11 adalah gambaran perancangan antarmuka halaman laporan. Untuk penjelasan dari objek antarmuka halaman laporan dapat dilihat pada Tabel 5.11

**Tabel 5.9 Keterangan Antarmuka Halaman Laporan**

No.	Nama Objek	Tipe	Keterangan
1	Page title	Text	Text untuk menampilkan judul halaman
2	Form rm	Form	Form yang digunakan untuk memasukkan data rm
3	Simpan	Button	Button yang digunakan untuk menyimpan data
4	nav	Nav tabs	Nav untuk menampilkan menu halaman laporan

## 5.2 Implementasi Sistem

Sub-bab implementasi sistem dilaksanakan setelah bab perancangan. Implementasi dilakukan sesuai dengan hasil yang diperoleh dari tahap analisis kebutuhan dan juga tahap perancangan. Setiap kebutuhan yang telah diidentifikasi dan ditemukan akan diimplementasikan ke dalam sistem. Implementasi mencakup struktur serta kode program berdasarkan kelas dan algoritma yang telah dibuat pada sub-bab perancangan. Pada sub-bab implementasi ini menjelaskan mengenai spesifikasi sistem, implementasi kode program, implementasi data, dan implementasi antarmuka.



### 5.2.1 Spesifikasi Sistem

Spesifikasi sistem yang digunakan dalam membangun sistem ini mencakup spesifikasi perangkat keras, perangkat lunak, sistem operasi, serta arsitektur sistem. Sub-bab berikut menjelaskan mengenai spesifikasi sistem yang digunakan untuk pembangunan perangkat lunak Aplikasi Rekam Medis RSIAPF.

**Tabel 5.10 Spesifikasi Perangkat Keras**

Nama Komponen	Spesifikasi
Laptop	-

**Tabel 5.11 Spesifikasi Perangkat Lunak**

Nama Komponen	Spesifikasi
Editor Dokumentasi	Microsoft Word 2016
Editor Perancangan	yEd, visual paradigm, pencil
Editor Pengkodean Program	Sublime Text 3
Bahasa Pemrograman	PHP, HTML, CSS, Javascript
Framework	Codeigniter
DBMS	PhpMyAdmin
Browser	MX5

**Tabel 5.12 Sistem Operasi**

Nama Komponen	Spesifikasi
Sistem Operasi Pembangunan	Windows 10
Sistem Operasi Server	Windows 10

### 5.2.2 Implementasi Kode Program

Sub sub-bab implementasi kode program dibuat berdasarkan perancangan komponen. Pada sub sub bab ini algoritma yang telah dibuat dalam bentuk *pseudocode* akan diterjemahkan menjadi kode bahasa pemrograman.

#### 5.2.2.1 Implementasi Kode Program *Method* simpan\_pendaftaran

Nama Class : pendaftaran.php

Nama Operasi : simpan\_pendaftaran()

Source code :

**Tabel 5.13 Implementasi Kode Program *Method* simpan\_pendaftaran**

1	<code>public function simpan_pendaftaran(){</code>
2	<code>    if (\$this-&gt;form_validation-&gt;run('form_pendaftaran')         == false){</code>
3	<code>        \$x['data'] = \$this-&gt;input-&gt;post('no_daftar');</code>

4	<code>\$this-&gt;load-</code>
	<code>&gt;view('V_pendaftaran/form_pendaftaran', \$x);</code>
5	<code>} else {</code>
6	<code>\$data = \$this-&gt;input-&gt;post();</code>
7	<code>\$insert = \$this-&gt;M_pendaftaran-</code>
	<code>&gt;simpan_pendaftaran(\$data);</code>
8	<code>if (\$insert) {</code>
9	<code>echo "&lt;script</code>
	<code>type=\"text/javascript\"&gt;alert('Data</code>
	<code>Berhasil Disimpan');&lt;/script&gt;";</code>
10	<code>} else {</code>
11	<code>echo "&lt;script</code>
	<code>type=\"text/javascript\"&gt;alert('Data Gagal</code>
	<code>Disimpan');&lt;/script&gt;";</code>
12	<code>}</code>
13	<code>redirect('pendaftaran');</code>
14	<code>}</code>
15	<code>}</code>

#### 5.2.2.2 Implementasi Kode Program *Method* `simpan_anc.php`

Nama Class : `anc.php`

Nama Operasi : `simpan_anc()`

Source code :

**Tabel 5.14 Implementasi Kode Program *Method* `simpan_anc`**

1	<code>public function simpan_anc(\$no_rm){</code>
2	<code>\$data['gambar'] =</code>
	<code>file_get_contents(\$_FILES['gambar']['tmp_name']);</code>
3	<code>\$("#gambar").change(function(){</code>
	<code>var fileInput =</code>
4	<code>document.getElementById('gambar');</code>
5	<code>var filePath = fileInput.value;</code>
6	<code>var allowedExtensions =</code>
	<code>/(\.jpg \.jpeg \.png \.gif)\$/i;</code>
7	<code>if(!allowedExtensions.exec(filePath)){</code>
8	<code>alert('File yang diupload hanya</code>
	<code>menggunakan ekstensi jpg, jpeg, png, dan</code>
	<code>gif');</code>

9	fileInput.value = '';
10	\$('#image_preview').html("")
11	return false;
12	}else{
13	var fsize = \$('#gambar')[0].files[0].size;
14	if(fsize>1048576) //do something if file size more than 1 mb (1048576)
15	{ alert('File yang diupload hanya menggunakan ekstensi jpg, jpeg, png, dan gif');
16	fileInput.value = '';
17	\$('#image_preview').html("")
18	return false;
19	}else{
20	if (fileInput.files && fileInput.files[0]) {
21	var reader = new FileReader();
22	reader.onload = function(e) {
23	\$('#image_preview').html("");
24	document.getElementById('image_preview };
25	reader.readAsDataURL(fileInput.files[0 ]);
26	}
27	}
28	}
29	});
30	}

### 5.2.2.3 Implementasi Kode Program *Method cetak\_persalinan*

Nama Class : laporan.php

Nama Operasi : cetak\_persalinan()

Source code :

**Tabel 5.15 Implementasi Kode Program *Method* cetak\_persalinan**

1	public function cetak_persalinan(){
2	\$data = \$this->M_laporan->show_persalinan(\$tgl1,
	\$ttl2);
3	\$no=1;
4	foreach (\$data as \$row){
5	\$pdf->Cell(10,6,\$no,1,0,'C');
6	\$pdf->Cell(40,6,\$row->nama,1,0);
7	\$pdf->Cell(20,6,\$row->umur,1,0);
8	\$pdf->Cell(40,6,date('d M Y', strtotime(\$row->
	lahir_tgl)),1,0);
9	\$pdf->Cell(20,6,\$row->jk_bayi,1,0, 'C');
10	\$pdf->Cell(30,6,\$row->bb,1,0);
11	\$pdf->Cell(30,6,\$row->pb,1,0);
12	\$pdf->Cell(30,6,\$row->alamat,1,1);
13	\$no++;
14	}
15	}

### 5.2.3 Implementasi Data

Pada implementasi data menggunakan *Data Definition Language* (DDL) merupakan bentuk fisik perancangan basis data yang sudah siap untuk diimplementasikan ke dalam database sehingga nama tabel yang ada pada DDL adalah merupakan nama asli dari tabel yang ada pada database. Tabel 5.16 berikut ini merupakan *Data Definition Language* (DDL) dari aplikasi rekam medis.

**Tabel 5.16 *Data Definition Language* Aplikasi Rekam Medis RSIAPF**

CREATE TABLE `anc` (
`id` int(15) NOT NULL AUTO_INCREMENT,
`no_rm` varchar(15) NOT NULL,
`nama` varchar(100) NOT NULL,
`tgl_periksa` date NOT NULL,
`berat_badan` varchar(100) NOT NULL,
`tinggi_badan` varchar(100) NOT NULL,
`tekanan_darah` varchar(100) NOT NULL,
`lila` varchar(100) NOT NULL,
`tinggi_fundus_uteri` varchar(100) NOT NULL,
`presentasi_janin` varchar(100) NOT NULL,
`denyut_jantung_janin` varchar(100) NOT NULL,
`tt` varchar(100) NOT NULL,
`tgl_tt` date NOT NULL,
`test_lab` text NOT NULL,
`terapiobat` varchar(100) NOT NULL,
`gambar` longblob NOT NULL,
`namadokter` varchar(100) NOT NULL,
`keterangan` text NOT NULL,
PRIMARY KEY (`id`),
KEY `no_rm` (`no_rm`),

```
KEY `no_rm_2` (`no_rm`)  
) ENGINE=InnoDB AUTO INCREMENT=10 DEFAULT CHARSET=latin1
```

Tabel *Data Definition Language* (DDL) diatas hanya ditampilkan 1 tabel saja, untuk data yang lengkap dapat dilihat pada Lampiran A *Data Definition Language* (DDL)

## 5.2.4 Implementasi Antarmuka

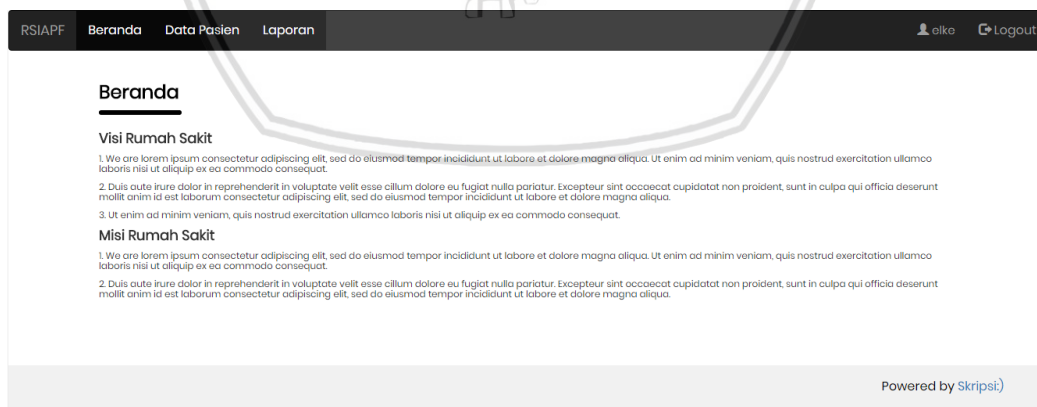
### 5.2.4.1 Implementasi Antarmuka Halaman *Login*



**Gambar 5.12 Implementasi Antarmuka Halaman Login**

Gambar 5.12 merupakan gambaran dari implementasi antarmuka halaman login sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

### 5.2.4.2 Implementasi Antarmuka Halaman Beranda



**Gambar 5.13 Implementasi Antarmuka Halaman Beranda**

Gambar 5.13 merupakan gambaran dari implementasi antarmuka halaman beranda sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

### 5.2.4.3 Implementasi Antarmuka Halaman Daftar Data Pasien

RSIAPF Beranda Data Pasien Laporan <span>elke Logout</span>						
Daftar Data Pasien						
<div>Tambah Data</div> <div> Show 10 entries <div>Search:</div> </div>						
No	No Pendaftaran	No RM	Nama	Tgl Lahir	Alergi	Aksi
1	1407180001	CP2345	Clara Puspa	09 Jan 1997	Dingin	<div>Ubat</div> <div>Edit</div>
2	1307180001	NA2121	Hana Anwar	19 Feb 1973	Udara dingin	<div>Ubat</div> <div>Edit</div>
Showing 1 to 2 of 2 entries				Previous	1	Next

Powered by Skripsi)

**Gambar 5.14 Implementasi Antarmuka Halaman Daftar Data Pasien**

Gambar 5.14 merupakan gambaran dari implementasi antarmuka halaman daftar data pasien sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

### 5.2.4.4 Implementasi Antarmuka Halaman Pendaftaran Pasien

RSIAPF Beranda Data Pasien Laporan <span>elke Logout</span>	
Pendaftaran Pasien	
No. Pendaftaran	1507180001
No. RM	No RM
Jenis	Umum
Nama Pasien	Nama Pasien
Jenis Kelamin	---
Agama	---
Tanggal Lahir	mm/dd/yyyy
Alamat	Alamat
Telepon	Telepon
Pekerjaan	Pekerjaan
Alergi	Alergi
Keluhan	Keluhan
<div>Simpan</div>	

**Gambar 5.15 Implementasi Antarmuka Halaman Pendaftaran Pasien**

Gambar 5.15 merupakan gambaran dari implementasi antarmuka halaman pendaftaran pasien sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.



#### 5.2.4.5 Implementasi Antarmuka Halaman Data Rekam Medis

The screenshot shows the RSIAPF application interface. At the top, there is a navigation bar with 'RSIAPF', 'Beranda', 'Data Pasien', and 'Laporan'. On the right, there are user icons for 'elke' and a 'Logout' button. Below the navigation bar, there is a horizontal menu with tabs: 'Ringkasan Masuk Keluar', 'Resume Medis', 'Rencana Pulang', 'A Kebidanan', 'A Keperawatan', 'MR3.1', 'MR3.2', 'Rawat Jalan', and 'Sebab Kematian'. The 'Ringkasan Masuk Keluar' tab is active. Below the tabs, there is a search bar with a dropdown for 'Kategori' and a 'Cari' button. A '+ Tambah Data' button is located above a table. The table has the following data:

No	Tanggal	Diagnosa Masuk	Nama Dokter	Action
1	01 Jul 2018	Komplikasi kehamilan	dr. Retno Budiati, Sp. OG	<a href="#">Buka</a> <a href="#">Edit</a>

**Gambar 5.16 Implementasi Antarmuka Halaman Data Rekam Medis**

Gambar 5.16 merupakan gambaran dari implementasi antarmuka halaman data rekam medis sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.6 Implementasi Antarmuka Halaman Tambah Data Ringkasan Masuk dan Keluar

The screenshot shows the 'Ringkasan Masuk Dan Keluar' form. It contains various input fields for patient data, organized into sections:

- Data Pribadi Pasien:** Includes fields for Tanggal (mm/dd/yyyy), No RM (NA2121), Tgl Lahir (1973-02-19), Nama Pasien (Hana Anwar), JK (P), Agama (Islam), Pekerjaan (PNS), Alamat (Jl. Muna 14/35 Makassar), and Telepon (08124259384).
- Status Perkawinan:** A dropdown menu.
- Cara Penerimaan:** A dropdown menu.
- Cara Masuk, Dikirim Oleh:** A dropdown menu.
- Bagi/Spes:** A dropdown menu.
- Ruang Rawat:** Includes fields for Ruang Rawat, Kamar/Kelas, and Lama Dirawat.
- Nama Penanggung Jawab Pembayaran:** A text field.
- Nama Keluarga:** A text field.
- Alamat Keluarga:** A text field.
- Tgl Masuk:** A date field (mm/dd/).
- Jam:** A time field (---:--).
- Pindah ke Ruang:** A text field.
- Tgl Pindah:** A date field (mm/dd/).
- Tgl Keluar:** A date field (mm/dd/).
- Jam Keluar:** A time field (---:--).
- Diagnosa Masuk:** A text field.

**Gambar 5.17 Potongan Antarmuka Halaman Tambah Data Ringkasan Masuk dan Keluar**

Gambar 5.17 merupakan potongan gambar dari implementasi antarmuka halaman tambah data ringkasan masuk dan keluar sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.7 Implementasi Antarmuka Halaman Tambah Data Resume Medis

**Gambar 5.18 Potongan Antarmuka Halaman Tambah Data Resume Medis**

Gambar 5.18 merupakan potongan gambar dari implementasi antarmuka halaman tambah data resume medis dan keluar sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.8 Implementasi Antarmuka Halaman Tambah Data Rencana Pulang

**Gambar 5.19 Potongan Antarmuka Halaman Tambah Data Rencana Pulang**

Gambar 5.19 merupakan potongan gambar dari implementasi antarmuka halaman tambah data rencana pulang sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.9 Implementasi Antarmuka Halaman Tambah Data Pengkajian Asuhan Kebidanan

Pengkajian Asuhan Kebidanan

---

Data Pasien

No RM CP2345	Ruang Rawat Ruang Rawat	Tanggal mm/dd/h	Jam --:--	JK P
Nama Lengkap Clara Puspita	Tgl Lahir 1997-01-11 Step 1	Umur Umur	Alamat Pandanwangi	
Agama Katolik	Rujukan --	Diagnosa Rujukan Diagnosa Rujukan		
Nama Keluarga Nama Keluarga	Telp Keluarga Telp Keluarga	Alamat Keluarga Alamat Keluarga		
Transportasi Waktu Datang --	Riwayat Alergi Riwayat Alergi	Dokter yang Memeriksa Nama Dokter	Nama Bidan Nama Bidan	

Anamnesa

Tanggal mm/dd/h	Jam --:--	Keluhan Utama Keluhan Utama
--------------------	--------------	--------------------------------

Riwayat Penyakit

**Gambar 5.20 Potongan Antarmuka Halaman Tambah Data Pengkajian Asuhan Kebidanan**

Gambar 5.20 merupakan potongan gambar dari implementasi antarmuka halaman tambah data pengkajian asuhan kebidanan sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.10 Implementasi Antarmuka Halaman Tambah Data Pengkajian Asuhan Keperawatan

Pengkajian Asuhan Keperawatan  
(dilengkapi dalam waktu 24 jam pertama pasien masuk ruang rawat)

---

Tgl Pengkajian mm/dd/yyyy	Jam Pengkajian --:--	Ruang Rawat Ruang Rawat
Nama Clara Puspita	Tgl Lahir 1997-01-09	JK P
Agama Katolik	Pendidikan --	Suku/Bangsa Suku/Bangsa
No RM CP2345	Sumber Data --	

Riwayat Kesehatan

Riwayat keluhan saat ini (Alasan masuk rumah sakit)

Alasan Masuk RS

Riwayat Opname --	Jika pernah opname, dengan sakit Sakit Sebelumnya	Di RS RS Opname Sebelumnya
BB Sebelum Sakit (kg) BB Sebelum Sakit (kg)	Riwayat Operasi --	Jika pernah operasi, di Tempat Operasi Sebelumnya

Kadaan Umum

**Gambar 5.21 Potongan Antarmuka Halaman Tambah Data Pengkajian Asuhan Keperawatan**

Gambar 5.21 merupakan potongan gambar dari implementasi antarmuka halaman tambah data pengkajian asuhan keperawatan sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.11 Implementasi Antarmuka Halaman Tambah Data MR3.1

Pemeriksaan Fisis Pertama

No RM Ibu CP2345	Tanggal mm/dd/yyyy	Nama Ibu Clara Puspita	Ruangan Ruangan
No RM Bayi No RM	JK -	Umur (bulan) Umur	BB (kg) BB (kg)
		PB (cm) PB (cm)	LLA (cm) LLA (cm)
Lingkar Kepala (cm) Lingkar Kepala (cm)	Lingkar Dada (cm) Lingkar Dada (cm)	Lingkar Perut (cm) Lingkar Perut (cm)	UG UG
PBI/UG PBI/UG	LK/UG LK/UG	Status Gizi Status Gizi	K.U Tekanan Darah
Nadi Nadi	Pernapasan Pernapasan	Suhu Suhu	Pucat Pucat
Sianosis Sianosis	Tunus Tunus	Ikterus Ikterus	
Turgor Turgor	Edema Edema	Kepala Kepala	Muka Muka
			Rambut Rambut
Ubun-ubun besar Ubun-ubun besar	Telinga Telinga	Mata Mata	Hidung Hidung
			Bibir Bibir
			Lidah Lidah

**Gambar 5.22 Potongan Antarmuka Halaman Tambah Data MR3.1**

Gambar 5.22 merupakan potongan gambar dari implementasi antarmuka halaman tambah data MR3.1 sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.12 Implementasi Antarmuka Halaman Tambah Data MR3.2

Pemeriksaan Fisis Dua

No RM Ibu CP2345	No Reg No Reg	Nama Ibu Clara Puspita	Lahir Tanggal mm/dd/yyyy	Nama Bayi Nama Bayi
Pukul --:--	Jenis Kelamin --	Jenis Kelahiran --		
Kondisi saat lahir --	Frekuensi Jantung --	Usaha Bernapas --	Tonus Otot --	
Refleksi --	Warna --			
Keterangan Keterangan penilaian setelah lahir				
Tindakan/Ventilasi				
<input type="checkbox"/> Pembersihan jalan nafas <input type="checkbox"/> Perangsangan <input type="checkbox"/> Pemberian O2 biasa (Frog Breathing) <input type="checkbox"/> Pemberian O2 dengan tekanan tidak langsung <input type="checkbox"/> Pemberian O2 dengan tekanan langsung <input type="checkbox"/> Penghisapan cairan lambung <input type="checkbox"/> Mouth to mouth breathing <input type="checkbox"/> Bag to mouth breathing <input type="checkbox"/> Mouth to tube breathing <input type="checkbox"/> Bag to tube breathing <input type="checkbox"/> External cardiac massage				

**Gambar 5.23 Potongan Antarmuka Halaman Tambah Data MR3.2**

Gambar 5.23 merupakan potongan gambar dari implementasi antarmuka halaman tambah data MR3.2 sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.13 Implementasi Antarmuka Halaman Tambah Data Rawat Jalan

Data Pemeriksaan ANC	
No RM	CP2345
Nama	Clara Puspa
Tanggal Periksa	mm/dd/yyyy
Berat Badan (kg)	Berat Badan (kg)
Tinggi Badan (cm)	Tinggi Badan (cm)
Tekanan Darah	Tekanan Darah
Lingkar Lengan Atas	Lingkar Lengan Atas
Tinggi Fundus Uteri	Tinggi Fundus Uteri
Presentasi Janin	Presentasi Janin
Denyut Jantung Janin	Denyut Jantung Janin
Imunisasi TT	--
Tanggal Imunisasi TT	mm/dd/yyyy

**Gambar 5.24 Potongan Antarmuka Halaman Tambah Data Rawat Jalan**

Gambar 5.24 merupakan potongan gambar dari implementasi antarmuka halaman tambah data rawat jalan sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

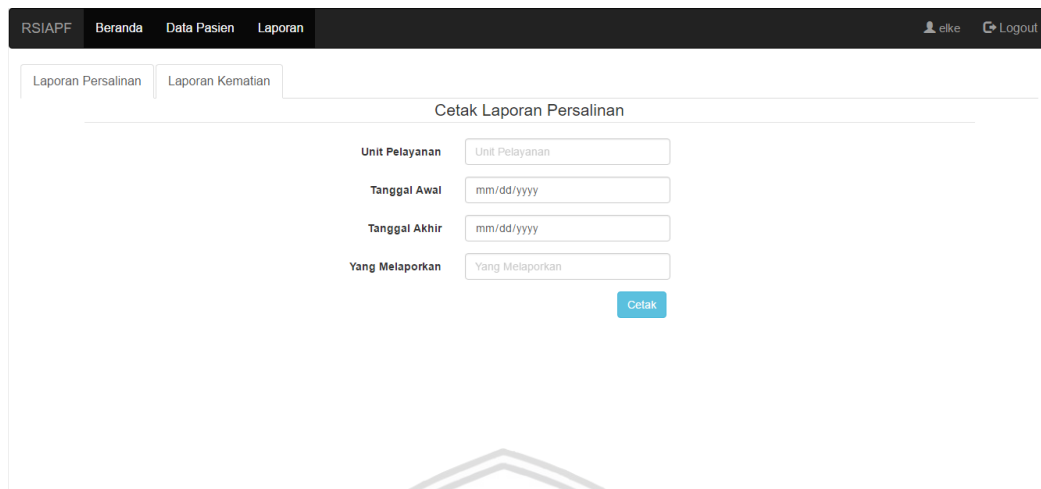
#### 5.2.4.14 Implementasi Antarmuka Halaman Tambah Data Sebab Kematian

Sebab Kematian	
No RM Pasien	Tanggal Kematian
CP2345	mm/dd/yyyy
a. Penyakit atau keadaan yang langsung mengakibatkan kematian	Akibat dari penyakit a
	Penyakit tersebut dalam bagian a disebabkan oleh (ak)
b. Penyakit (bila ada) yang menjadi lantaran timbulnya sebab kematian tersebut pada a dengan menyebut penyakit yang menjadi pokok pangkal	Akibat dari penyakit b
	Penyakit tersebut dalam bagian b disebabkan oleh (ak)
Akibat dari penyakit c (yang lainnya)	Penyakit lain
Lain-lain	Disamping penyakit-penyakit tersebut diatas terdapat
Penyakit-penyakit lain yang berarti dan mempengaruhi pula kematian ini, tetapi tidak ada hubungan dengan penyakit-penyakit tersebut dalam bagian a, b, dan c	
Lamanya mulai sakit hingga meninggal dunia	
Lamanya mulai sakit hingga meninggal dunia	
KETERANGAN KHUSUS UNTUK	
I. MATI KARENA RUDA PAKSA (Violent death)	
a. Macam Ruda Paksa	

**Gambar 5.25 Potongan Antarmuka Halaman Tambah Data Sebab Kematian**

Gambar 5.25 merupakan potongan gambar dari implementasi antarmuka halaman tambah data sebab kematian sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.

#### 5.2.4.15 Implementasi Antarmuka Halaman Laporan



RSIAPF Beranda Data Pasien Laporan

elke Logout

Laporan Persalinan Laporan Kematian

Cetak Laporan Persalinan

Unit Pelayanan

Tanggal Awal

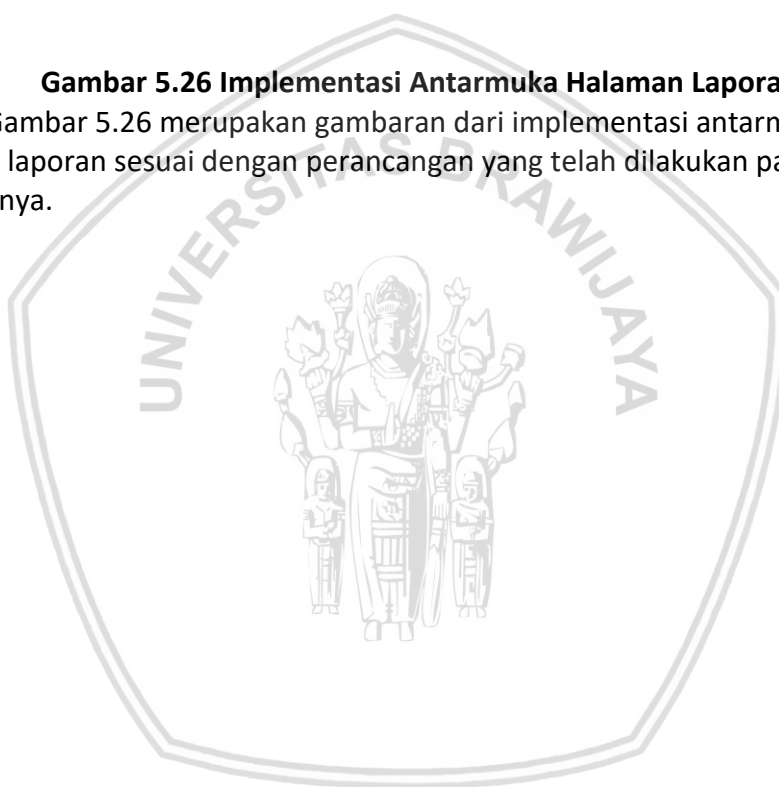
Tanggal Akhir

Yang Melaporkan

Cetak

**Gambar 5.26 Implementasi Antarmuka Halaman Laporan**

Gambar 5.26 merupakan gambaran dari implementasi antarmuka halaman laporan sesuai dengan perancangan yang telah dilakukan pada sub-bab sebelumnya.





## BAB 6 PENGUJIAN SISTEM

Pada bab ini akan dibahas mengenai pengujian sistem pada Aplikasi Rekam Medis RSIAPF setelah tahap implementasi. Tahap pengujian ini dilakukan guna mengetahui apakah implementasi sistem sudah sesuai dengan hasil analisis kebutuhan dan perancangan sistem. Pengujian yang dilakukan adalah pengujian unit, pengujian integrasi, pengujian validasi, serta pengujian *compatibility*.

### 6.1 Pengujian Unit

Pengujian unit digunakan untuk menguji setiap unit dari sistem komponen, kelas, maupun objek. Pengujian unit merupakan pengujian *white-box* dengan menggunakan *basis path testing*.

#### 6.1.1 Pengujian Unit *Method* `simpan_pendaftaran`

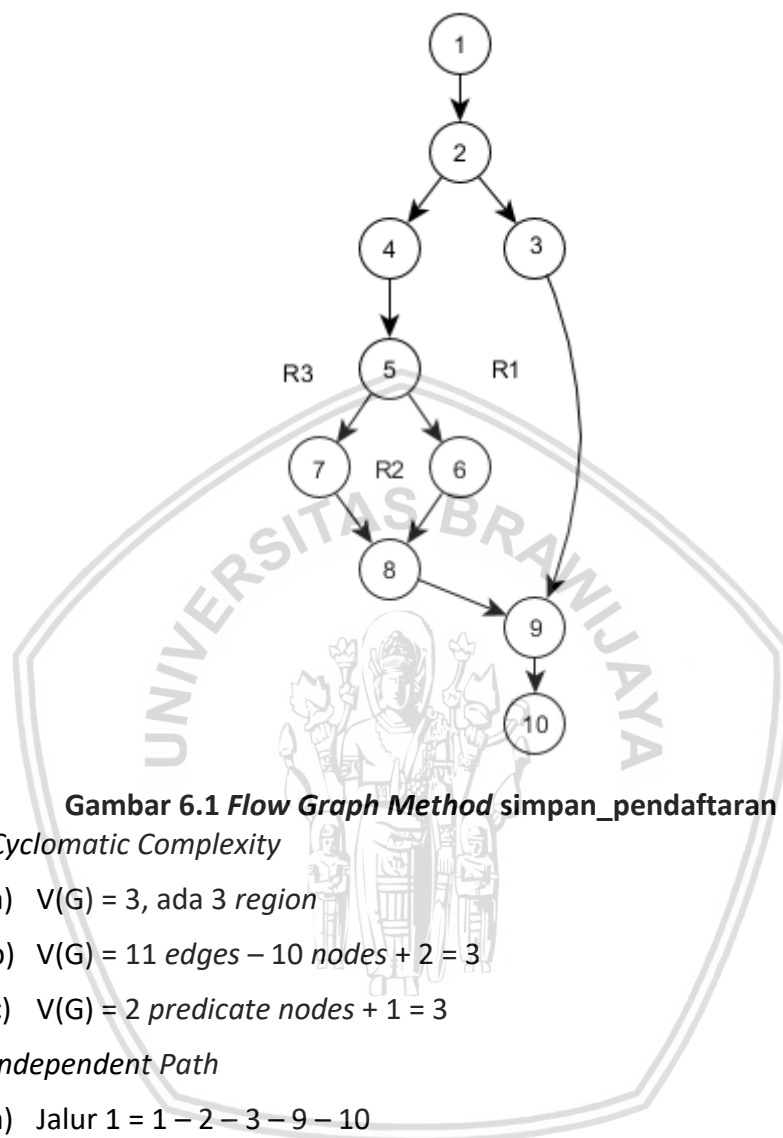
##### 1. Pseudocode

**Tabel 6.1 Pseudocode Method `simpan_pendaftaran`**

1	mulai method <code>simpan_pendaftaran</code> (1)	
2	if <code>form_pendaftaran = false</code> (2)	
3	masukkan data berdasarkan <code>no_daftar</code>	} (3)
4	tampil view <code>form_pendaftaran</code>	
5	else	} (4)
6	masukkan data	
7	memanggil method <code>simpan_pendaftaran</code> dari kelas <code>M_pendaftaran</code>	
8	if ( <code>\$insert</code> ) (5)	
9	cetak data berhasil disimpan (6)	} (7)
10	else	
11	cetak data gagal disimpan	
12	end if (8)	
13	redirect('pendaftaran') } (9)	
14	end if	
15	end (10)	

## 2. Basis Path Testing

### a. Flow Graph



**Gambar 6.1 Flow Graph Method simpan\_pendaftaran**

### b. Cyclomatic Complexity

- a)  $V(G) = 3$ , ada 3 region
- b)  $V(G) = 11 \text{ edges} - 10 \text{ nodes} + 2 = 3$
- c)  $V(G) = 2 \text{ predicate nodes} + 1 = 3$

### c. Independent Path

- a) Jalur 1 = 1 – 2 – 3 – 9 – 10
- b) Jalur 2 = 1 – 2 – 4 – 5 – 6 – 8 – 9 – 10
- c) Jalur 3 = 1 – 2 – 3 – 5 – 7 – 8 – 9 – 10

**Tabel 6.2 Hasil Pengujian Unit Method simpan\_pendaftaran**

No.	No Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Nilai form_pendaftaran = false	Menampilkan form pendaftaran	Menampilkan form pendaftaran	Valid
2	2	Nilai form_pendaftaran = true, No RM =	Menampilkan pesan data	Menampilkan pesan data	Valid

		CP1231, nama = Clara Pusparani	berhasil disimpan	berhasil disimpan	
3	3	form_pendaftaran = true, <i>field</i> nama = kosong	Menampilkan pesan data gagal disimpan	Menampilkan pesan data gagal disimpan	Valid

## 6.1.2 Pengujian Unit *Method* *simpan\_anc*

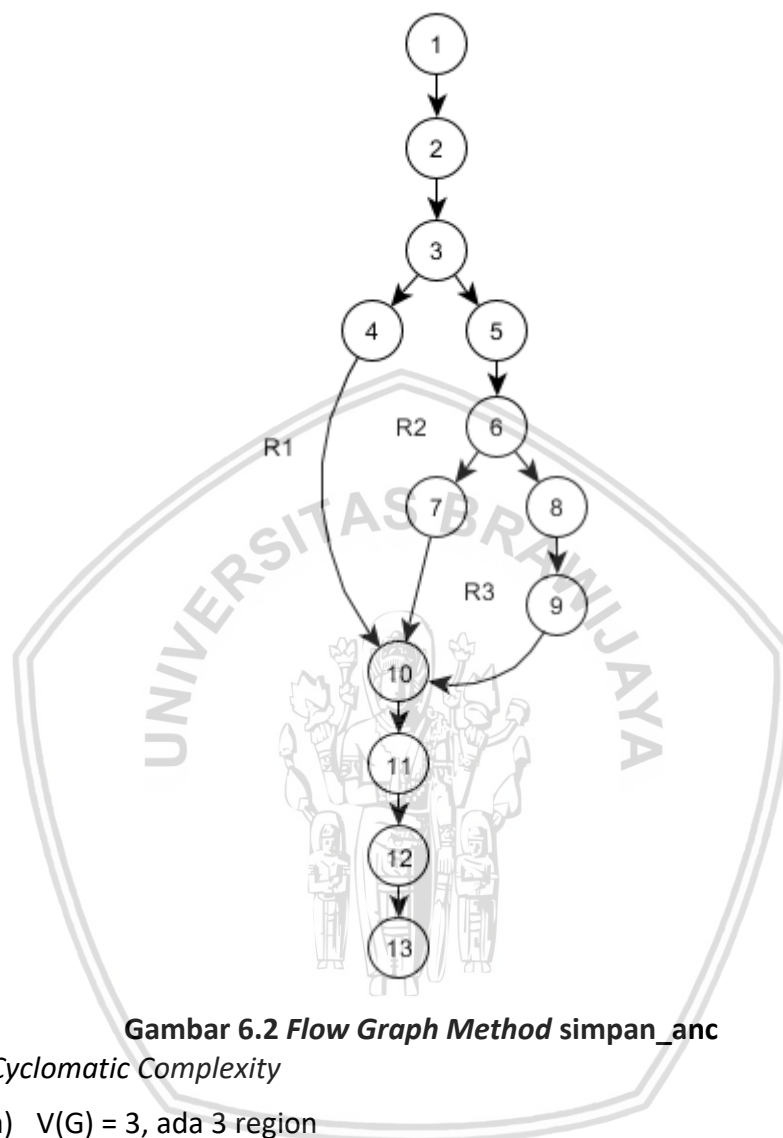
### 1. Pseudocode

**Tabel 6.3 Pseudocode Method *simpan\_anc***

1	mulai method <i>simpan_anc</i> (1)	
2	pilih dan ambil data gambar	
3	ambil value <i>fileInput</i> melalui id	
4	ambil value <i>filePath</i> dari value <i>fileInput</i>	} (2)
5	atur ekstensi file	
6	if <i>!allowedExtensions.exec(filePath)</i> (3)	
7	tampil "File yang diupload hanya menggunakan ekstensi jpg, jpeg, png, dan gif"	} (4)
8	set value <i>fileInput</i>	
9	tampil tinjauan gambar di html	
10	return false	
11	else	
12	ambil <i>fsize</i> gambar } (5)	
13	if ( <i>fsize</i> >1048576) (6)	
14	tampil "File yang diupload lebih dari 1MB"	} (7)
15	set value <i>fileInput</i>	
16	tampil tinjauan gambar di html	
17	return false	
18	else	
19	if ( <i>fileInput.files</i> && <i>fileInput.files</i> [0]) } (8)	
20	tampil tinjauan gambar di html (9)	
21	end if (10)	
22	end if (11)	
23	end if (12)	
24	end (13)	

## 2. Basis Path Testing

### a. Flow Graph



**Gambar 6.2 Flow Graph Method simpan\_anc**

### b. Cyclomatic Complexity

- $V(G) = 3$ , ada 3 region
- $V(G) = 14 \text{ edges} - 13 \text{ nodes} + 2 = 3$
- $V(G) = 2 \text{ predicate nodes} + 1 = 3$

### c. Independent Path

- Jalur 1 = 1 – 2 – 3 – 4 – 10 – 11 – 12 – 13
- Jalur 2 = 1 – 2 – 3 – 5 – 6 – 7 – 10 – 11 – 12 – 13
- Jalur 3 = 1 – 2 – 3 – 5 – 6 – 8 – 9 – 10 – 11 – 12 – 13

**Tabel 6.4 Hasil Pengujian Unit *Method* simpan\_anc**

No.	No Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	File dengan ekstensi .docx	Menampilkan pesan file yang diupload hanya menggunakan ekstensi jpg, jpeg, png, dan gif	Menampilkan pesan file yang diupload hanya menggunakan ekstensi jpg, jpeg, png, dan gif	Valid
2	2	File dengan ekstensi .jpg, ukuran file 1,8MB	Menampilkan pesan file yang diupload lebih dari 1MB	Menampilkan pesan file yang diupload lebih dari 1MB	Valid
3	3	File dengan ekstensi .jpg, ukuran file 120kb	Menampilkan pratinjau gambar	Menampilkan pratinjau gambar	Valid

### 6.1.3 Pengujian Unit *Method* cetak\_persalinan

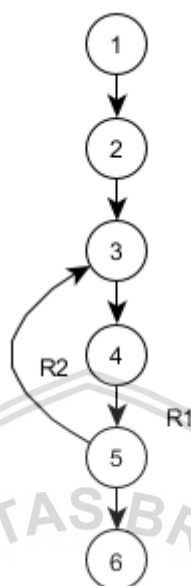
#### 1. Pseudocode

**Tabel 6.5 Pseudocode *Method* cetak\_persalinan**

1	mulai method cetak_persalinan (1)	
2	memanggil method show_persalinan dari kelas M_laporan untuk mengambil data diantara tgl1 dan tgl2	(2)
3	no = 1	
4	foreach row (3)	
5	no = ambil data no	
6	nama = ambil data nama	
7	umur = ambil data umur	
8	lahir_tgl = ambil data tanggal lahir	(4)
9	jk_bayi = ambil data jenis kelamin bayi	
10	bb = ambil data berat badan bayi	
11	pb = ambil data panjang badan bayi	
12	alamat = ambil data alamat	
13	increment no	
14	end foreach (5)	
15	end (6)	

## 2. Basis Path Testing

### a. Flow Graph



**Gambar 6.3 Flow Graph Method cetak\_persalinan**

### b. Cyclomatic Complexity

- a)  $V(G) = 2$ , ada 2 region
- b)  $V(G) = 7 \text{ edges} - 7 \text{ nodes} + 2 = 2$
- c)  $V(G) = 1 \text{ predicate nodes} + 1 = 2$

### c. Independent Path

- a) Jalur 1 = 1 – 2 – 3 – 4 – 5 – 6
- b) Jalur 1 = 1 – 2 – 3 – 4 – 5 – 3 – 4 – 5 – 6

**Tabel 6.6 Hasil Pengujian Unit Method cetak\_persalinan**

No.	No Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	tgl 1 = 01/07/18, tgl 2 = 01/08/18	Menampilkan data	Menampilkan data	Valid
2	2	tgl 1 = 08/07/18, tgl 2 = 01/07/18	Melakukan perulangan dan mengambil data	Melakukan perulangan dan mengambil data	Valid



## 6.2 Pengujian Validasi

Pengujian validasi dilaksanakan untuk memastikan bahwa sistem yang dibangun sudah sesuai dengan kebutuhan yang telah dijelaskan pada bab sebelumnya. Pengujian ini merupakan pengujian *black-box* dengan jenis *scenario-based testing*.

### 6.2.1 Pengujian Validasi Login

#### 1. Kasus Uji *Login*

**Tabel 6.7 Kasus Uji *Login***

<b>Nama Kasus Uji</b>	Kasus Uji <i>Login</i> sebagai
<b>Kode Kebutuhan</b>	RM_F0100
<b>Prosedur</b>	1. Memasukkan <i>username</i> = elke 2. Memasukkan <i>password</i> = admin 3. Menekan tombol <i>login</i>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman utama
<b>Hasil</b>	Sistem berhasil menampilkan halaman utama
<b>Status</b>	Valid

#### 2. Kasus Uji *Login* Alternatif 1

**Tabel 6.8 Kasus Uji *Login* Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji <i>Login</i> sebagai <i>User</i>
<b>Kode Kebutuhan</b>	RM_F0100
<b>Prosedur</b>	1. Memasukkan <i>username</i> = elke 2. Menekan tombol <i>login</i>
<b>Hasil yang diharapkan</b>	Sistem akan memberikan peringatan untuk mengisi <i>field</i> yang kosong
<b>Hasil</b>	Sistem berhasil memberikan peringatan untuk mengisi <i>field</i> yang kosong
<b>Status</b>	Valid

#### 3. Kasus Uji *Login* Alternatif 2

**Tabel 6.9 Kasus Uji *Login* Alternatif 2**

<b>Nama Kasus Uji</b>	Kasus Uji <i>Login</i> sebagai <i>User</i>
<b>Kode Kebutuhan</b>	RM_F0100
<b>Prosedur</b>	1. Memasukkan <i>username</i> = nadia 2. Memasukkan <i>password</i> = admin 3. Menekan tombol <i>login</i>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan pesan <i>username</i> dan <i>password</i> salah
<b>Hasil</b>	Sistem berhasil menampilkan pesan <i>username</i> dan <i>password</i> salah

Status	Valid
--------	-------

## 6.2.2 Pengujian Validasi Logout

### 1. Kasus Uji Logout

**Tabel 6.10 Kasus Uji Logout**

Nama Kasus Uji	Kasus Uji Logout
Kode Kebutuhan	RM_F0200
Prosedur	1. Login sebagai Perekam Medis 2. Menekan tombol logout
Hasil yang diharapkan	Sistem akan menampilkan halaman login
Hasil	Sistem menampilkan halaman login
Status	Valid

## 6.2.3 Pengujian Validasi Tambah Data Pendaftaran Pasien

### 1. Kasus Uji Tambah Data Pendaftaran Pasien

**Tabel 6.11 Kasus Uji Tambah Data Pendaftaran Pasien**

Nama Kasus Uji	Kasus Uji Tambah Data Pendaftaran Pasien
Kode Kebutuhan	RM_F0400
Prosedur	1. Mengakses halaman pendaftaran pasien 2. Memasukkan NO RM = NA2121 3. Memasukkan jenis = Umum 4. Memasukkan nama = Hana Anwar 5. Memasukkan jenis kelamin = P 6. Memasukkan agama = Islam 7. Memasukkan tanggal lahir = 19/02/1973 8. Memasukkan alamat = Jl. Muna 14/35 Makassar 9. Memasukkan no telp = 08124259384 10. Memasukkan pekerjaan = PNS 11. Memasukkan alergi = Udara Dingin 12. Menekan tombol simpan
Hasil yang diharapkan	Sistem akan menyimpan data pendaftaran pasien
Hasil	Sistem berhasil menyimpan data pendaftaran pasien
Status	Valid

### 2. Kasus Uji Tambah Data Pendaftaran Pasien Alternatif 1

**Tabel 6.12 Kasus Uji Tambah Data Pendaftaran Pasien Alternatif 1**

Nama Kasus Uji	Kasus Uji Tambah Data Pendaftaran Pasien
Kode Kebutuhan	RM_F0400
Prosedur	1. Mengakses halaman pendaftaran pasien

	2. Memasukkan NO RM = NA2121 3. Memasukkan jenis = Umum 4. Memasukkan nama = Hana Anwar 5. Memasukkan jenis kelamin = P 6. Memasukkan agama = Islam 7. Tidak memasukkan tanggal lahir 8. Memasukkan alamat = Jl. Muna 14/35 Makassar 9. Memasukkan no telp = 08124259384 10. Memasukkan pekerjaan = PNS 11. Memasukkan alergi = Udara Dingin 12. Menekan tombol Simpan
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang kosong
<b>Hasil</b>	Sistem menampilkan peringatan untuk mengisi <i>field</i> yang kosong
<b>Status</b>	Valid

#### 6.2.4 Pengujian Validasi Lihat Daftar Data Pasien

1. Kasus Uji Lihat Daftar Data Pasien

**Tabel 6.13 Kasus Uji Lihat Daftar Data Pasien**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Daftar Data Pasien
<b>Kode Kebutuhan</b>	RM_F0300
<b>Prosedur</b>	1. <i>Login</i> sebagai <i>Perekam Medis</i> 2. Mengakses halaman daftar data pasien
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman daftar data pasien
<b>Hasil</b>	Sistem menampilkan halaman daftar data pasien
<b>Status</b>	Valid

#### 6.2.5 Pengujian Validasi Edit Data Pendaftaran Pasien

1. Kasus Uji Edit Data Pendaftaran Pasien

**Tabel 6.14 Kasus Uji Edit Data Pendaftaran Pasien**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Pendaftaran Pasien
<b>Kode Kebutuhan</b>	RM_F0500
<b>Prosedur</b>	1. Mengakses halaman pendaftaran pasien 2. Menekan tombol edit 3. Melakukan <i>update</i> alamat = Jl. Dobu no. 4 Makassar 4. Melakukan <i>update</i> alergi = Udara Dingin, Debu 5. Menekan tombol <i>Update</i>

<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data pendaftaran yang telah di <i>update</i> dan menampilkan kembali halaman daftar data pasien
<b>Hasil</b>	Sistem berhasil menyimpan data pendaftaran yang telah di <i>update</i> dan menampilkan kembali halaman daftar data pasien
<b>Status</b>	Valid

## 2. Kasus Uji Edit Data Pendaftaran Pasien Alternatif 1

**Tabel 6.15 Kasus Uji Edit Data Pendaftaran Pasien Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Pendaftaran Pasien Alternatif 1
<b>Kode Kebutuhan</b>	RM_F0500
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman pendaftaran pasien</li> <li>2. Menekan tombol edit</li> <li>3. Melakukan <i>update</i> alamat = Jl. Doju no. 4 Makassar</li> <li>4. Mengosongkan <i>field</i> alergi</li> <li>6. Menekan tombol <i>Update</i></li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang kosong
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang kosong
<b>Status</b>	Valid

## 6.2.6 Pengujian Validasi Cari Data Pasien

### 1. Kasus Uji Cari Data Pasien

**Tabel 6.16 Kasus Uji Cari Data Pasien**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Pasien
<b>Kode Kebutuhan</b>	RM_F0600
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman pendaftaran pasien</li> <li>2. Memasukkan <i>keyword</i> = dingin</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang memiliki nama Hana Anwar
<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang memiliki nama Hana Anwar
<b>Status</b>	Valid

## 2. Kasus Uji Cari Data Pasien Alternatif 1

**Tabel 6.17 Kasus Uji Cari Data Pasien Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Pasien Alternatif 1
<b>Kode Kebutuhan</b>	RM_F0600
<b>Prosedur</b>	1. Mengakses halaman pendaftaran pasien 2. Memasukkan <i>keyword</i> = angin
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan <i>no matching records found</i>
<b>Hasil</b>	Sistem berhasil menampilkan <i>no matching records found</i>
<b>Status</b>	Valid

**6.2.7 Pengujian Validasi Tambah Data Ringkasan Masuk dan Keluar**

## 1. Kasus Uji Tambah Data Ringkasan Masuk dan Keluar

**Tabel 6.18 Kasus Uji Tambah Data Ringkasan Masuk dan Keluar**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Ringkasan Masuk dan Keluar
<b>Kode Kebutuhan</b>	RM_F0700
<b>Prosedur</b>	1. Mengakses halaman Ringkasan Masuk dan Keluar 2. Menekan tombol Tambah Data 3. Memasukkan tanggal = 07/15/2018 4. Memasukkan status perkawinan = Kawin 5. Memasukkan cara penerimaan = IGD 6. Memasukkan cara masuk = Dokter 7. Memasukkan bag/spes = Obgyn 8. Memasukkan riwayat imunisasi = BCG, Polio 9. Memasukkan nama dokter = dr. Dedi C, Sp.OG 10. Menekan tombol simpan
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data ringkasan masuk dan keluar dan menampilkan halaman data ringkasan masuk dan keluar
<b>Hasil</b>	Sistem berhasil menyimpan data ringkasan masuk dan keluar dan menampilkan halaman data ringkasan masuk dan keluar
<b>Status</b>	Valid

## 2. Kasus Uji Tambah Data Ringkasan Masuk dan Keluar Alternatif 1

Tabel 6.19 Kasus Uji Tambah Data Ringkasan Masuk dan Keluar Alternatif 1

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Ringkasan Masuk dan Keluar Alternatif 1
<b>Kode Kebutuhan</b>	RM_F0700
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Ringkasan Masuk dan Keluar</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Tidak memasukkan tanggal</li> <li>4. Memasukkan status perkawinan = Kawin</li> <li>5. Memasukkan cara penerimaan = IGD</li> <li>6. Memasukkan cara masuk = Dokter</li> <li>7. Memasukkan bag/spes = Obgyn</li> <li>8. Menekan tombol simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field yang required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field yang required</i>
<b>Status</b>	Valid

## 6.2.8 Pengujian Validasi Lihat Data Ringkasan Masuk dan Keluar

## 1. Kasus Uji Lihat Data Ringkasan Masuk dan Keluar

Tabel 6.20 Kasus Uji Lihat Data Ringkasan Masuk dan Keluar

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Ringkasan Masuk dan Keluar
<b>Kode Kebutuhan</b>	RM_F0800
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. <i>Login sebagai Perekam Medis</i></li> <li>2. Mengakses halaman Ringkasan Masuk dan Keluar</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data ringkasan masuk dan keluar
<b>Hasil</b>	Sistem menampilkan halaman data ringkasan masuk dan keluar
<b>Status</b>	Valid

## 6.2.9 Pengujian Validasi Lihat Detail Data Ringkasan Masuk dan Keluar

## 1. Kasus Uji Lihat Detail Data Ringkasan Masuk dan Keluar

Tabel 6.21 Kasus Uji Lihat Detail Data Ringkasan Masuk dan Keluar

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data Ringkasan Masuk dan Keluar
<b>Kode Kebutuhan</b>	RM_F0900
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. <i>Login sebagai Perekam Medis</i></li> </ol>



	2. Mengakses halaman Ringkasan Masuk dan Keluar 3. Menekan tombol Buka
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data ringkasan masuk dan keluar
<b>Hasil</b>	Sistem menampilkan detail data ringkasan masuk dan keluar
<b>Status</b>	Valid

#### 6.2.10 Pengujian Validasi Edit Data Ringkasan Masuk dan Keluar

##### 1. Kasus Uji Edit Data Ringkasan Masuk dan Keluar

**Tabel 6.22 Kasus Uji Edit Data Ringkasan Masuk dan Keluar**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Ringkasan Masuk dan Keluar
<b>Kode Kebutuhan</b>	RM_F1000
<b>Prosedur</b>	1. Mengakses halaman Ringkasan Masuk dan Keluar 2. Menekan tombol edit 3. Melakukan <i>update</i> tanggal = 12/01/2017 4. Melakukan <i>update</i> penanggungjawab pembayaran = Dita 5. Menekan tombol <i>Update</i>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data ringkasan masuk dan keluar yang telah di <i>update</i> dan menampilkan kembali halaman data ringkasan masuk dan keluar
<b>Hasil</b>	Sistem berhasil menyimpan data ringkasan masuk dan keluar yang telah di <i>update</i> dan menampilkan kembali halaman data ringkasan masuk dan keluar
<b>Status</b>	Valid

#### 6.2.11 Pengujian Validasi Cari Data Ringkasan Masuk dan Keluar

##### 1. Kasus Uji Cari Data Ringkasan Masuk dan Keluar

**Tabel 6.23 Kasus Uji Cari Data Ringkasan Masuk dan Keluar**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Ringkasan Masuk dan Keluar
<b>Kode Kebutuhan</b>	RM_F1100
<b>Prosedur</b>	1. Mengakses halaman Ringkasan Masuk dan Keluar 2. Memasukkan <i>keyword</i> = de 3. Memasukkan kategori pencarian = Nama Dokter

	4. Menekan tombol Cari
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang mempunyai Nama Dokter yang memiliki <i>value</i> de
<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang mempunyai Nama Dokter yang memiliki <i>value</i> de
<b>Status</b>	Valid

### 6.2.12 Pengujian Validasi Tambah Data Resume Medis

#### 1. Kasus Uji Tambah Data Resume Medis

**Tabel 6.24 Kasus Uji Tambah Data Resume Medis**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Resume Medis
<b>Kode Kebutuhan</b>	RM_F1200
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Resume Medis</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Memasukkan tanggal masuk = 07/15/2018</li> <li>4. Memasukkan umur = 22</li> <li>5. Memasukkan penanggungjawab pembayaran = Lala</li> <li>6. Memasukkan diagnosa masuk = Jantung</li> <li>7. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data resume medis dan menampilkan halaman kembali data resume medis
<b>Hasil</b>	Sistem berhasil menyimpan data resume medis dan menampilkan halaman kembali data resume medis
<b>Status</b>	Valid

#### 2. Kasus Uji Tambah Data Resume Medis Alternatif 1

**Tabel 6.25 Kasus Uji Tambah Data Resume Medis Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Resume Medis Alternatif 1
<b>Kode Kebutuhan</b>	RM_F1200
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Resume Medis</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Mengosongkan tanggal masuk</li> <li>4. Memasukkan umur = 22</li> <li>5. Memasukkan penanggungjawab pembayaran = Dina</li> <li>6. Memasukkan diagnosa masuk = Jantung</li> <li>7. Menekan tombol Simpan</li> </ol>

<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

### 6.2.13 Pengujian Validasi Lihat Data Resume Medis

1. Kasus Uji Lihat Data Resume Medis

**Tabel 6.26 Kasus Uji Lihat Data Resume Medis**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Resume Medis
<b>Kode Kebutuhan</b>	RM_F1300
<b>Prosedur</b>	1. <i>Login sebagai Perekam Medis</i> 2. Mengakses halaman Resume Medis
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data resume medis
<b>Hasil</b>	Sistem menampilkan halaman data resume medis
<b>Status</b>	Valid

### 6.2.14 Pengujian Validasi Lihat Detail Data Resume Medis

1. Kasus Uji Lihat Detail Data Resume Medis

**Tabel 6.27 Kasus Uji Lihat Detail Data Resume Medis**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data Resume Medis
<b>Kode Kebutuhan</b>	RM_F1400
<b>Prosedur</b>	1. <i>Login sebagai Perekam Medis</i> 2. Mengakses halaman Resume Medis 3. Menekan tombol Buka
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data resume medis
<b>Hasil</b>	Sistem menampilkan detail data resume medis
<b>Status</b>	Valid

### 6.2.15 Pengujian Validasi Edit Data Resume Medis

1. Kasus Uji Edit Data Resume Medis

**Tabel 6.28 Kasus Uji Edit Data Resume Medis**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Resume Medis
<b>Kode Kebutuhan</b>	RM_F1500
<b>Prosedur</b>	1. Mengakses halaman Resume Medis 2. Menekan tombol edit

	3. Melakukan <i>update</i> ringkasan riwayat penyakit = pasien menderita penyakit jantung yang bersamaan dengan munculnya penyakit empedu 4. Melakukan <i>update</i> anjuran = istirahat cukup, tidur harus minimal 8 jam 5. Menekan tombol <i>Update</i>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data resume medis yang telah di <i>update</i> dan menampilkan kembali halaman data resume medis
<b>Hasil</b>	Sistem berhasil menyimpan data resume medis yang telah di <i>update</i> dan menampilkan kembali halaman data resume medis
<b>Status</b>	Valid

### 6.2.16 Pengujian Validasi Cari Data Resume Medis

#### 1. Kasus Uji Cari Data Resume Medis

**Tabel 6.29 Kasus Uji Cari Data Resume Medis**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Resume Medis
<b>Kode Kebutuhan</b>	RM_F1600
<b>Prosedur</b>	1. Mengakses halaman Resume Medis 2. Memasukkan <i>keyword</i> = prek 3. Memasukkan kategori pencarian = Diagnosa Utama 4. Menekan tombol Cari
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang mempunyai Diagnosa Utama yang memiliki <i>value</i> prek
<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang mempunyai Diagnosa Utama yang memiliki <i>value</i> prek
<b>Status</b>	Valid

### 6.2.17 Pengujian Validasi Tambah Data Rencana Pulang

#### 1. Kasus Uji Tambah Data Rencana Pulang

**Tabel 6.30 Kasus Uji Tambah Data Rencana Pulang**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Rencana Pulang
<b>Kode Kebutuhan</b>	RM_F1700
<b>Prosedur</b>	1. Mengakses halaman Rencana Pulang 2. Menekan tombol Tambah Data 3. Memasukkan tanggal = 07/15/2018

	4. Memasukkan dokter yang merawat = dr. Alif T, Sp. OG 5. Memasukkan diagnosa keluar = Preklamsia 6. Memasukkan hasil pemeriksaan = USG, KMS 7. Menekan tombol Simpan
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data rencana pulang dan menampilkan halaman kembali data rencana pulang
<b>Hasil</b>	Sistem berhasil menyimpan data rencana pulang dan menampilkan halaman kembali data rencana pulang
<b>Status</b>	Valid

## 2. Kasus Uji Tambah Data Resume Medis Alternatif 1

**Tabel 6.31 Kasus Uji Tambah Data Resume Medis Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Resume Medis Alternatif 1
<b>Kode Kebutuhan</b>	RM_F1700
<b>Prosedur</b>	1. Mengakses halaman Rencana Pulang 2. Menekan tombol Tambah Data 3. Mengosongkan tanggal 4. Memasukkan dokter yang merawat = dr. Alif T, Sp. OG 5. Mengosongkan diagnosa keluar 6. Memasukkan hasil pemeriksaan = USG, KMS 7. Menekan tombol Simpan
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field yang required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field yang required</i>
<b>Status</b>	Valid

## 6.2.18 Pengujian Validasi Lihat Data Rencana Pulang

### 1. Kasus Uji Lihat Data Rencana Pulang

**Tabel 6.32 Kasus Uji Lihat Data Rencana Pulang**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Rencana Pulang
<b>Kode Kebutuhan</b>	RM_F1800
<b>Prosedur</b>	1. <i>Login sebagai Perekam Medis</i> 2. Mengakses halaman Rencana Pulang

<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data rencana pulang
<b>Hasil</b>	Sistem menampilkan halaman data rencana pulang
<b>Status</b>	Valid

### 6.2.19 Pengujian Validasi Lihat Detail Data Rencana Pulang

1. Kasus Uji Lihat Detail Data Rencana Pulang

**Tabel 6.33 Kasus Uji Lihat Detail Data Rencana Pulang**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data Rencana Pulang
<b>Kode Kebutuhan</b>	RM_F1900
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. <i>Login</i> sebagai <i>Perekam Medis</i></li> <li>2. Mengakses halaman Rencana Pulang</li> <li>3. Menekan tombol Buka</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data rencana pulang
<b>Hasil</b>	Sistem menampilkan detail data rencana pulang
<b>Status</b>	Valid

### 6.2.20 Pengujian Validasi Edit Data Rencana Pulang

1. Kasus Uji Edit Data Rencana Pulang

**Tabel 6.34 Kasus Uji Edit Data Rencana Pulang**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Rencana Pulang
<b>Kode Kebutuhan</b>	RM_F2000
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Rencana Pulang</li> <li>2. Menekan tombol edit</li> <li>3. Melakukan <i>update</i> aktifitas dan istirahat = kurangi kegiatan diluar ruangan</li> <li>4. Melakukan <i>update</i> TD = 120/70</li> <li>5. Melakukan <i>update</i> suhu tubuh = 38°C</li> <li>6. Menekan tombol <i>Update</i></li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data rencana pulang yang telah di <i>update</i> dan menampilkan kembali halaman data rencana pulang
<b>Hasil</b>	Sistem berhasil menyimpan data rencana pulang yang telah di <i>update</i> dan menampilkan kembali halaman data rencana pulang
<b>Status</b>	Valid



### 6.2.21 Pengujian Validasi Cari Data Rencana Pulang

#### 1. Kasus Uji Cari Data Rencana Pulang

**Tabel 6.35 Kasus Uji Cari Data Rencana Pulang**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Rencana Pulang
<b>Kode Kebutuhan</b>	RM_F2100
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Rencana Pulang</li> <li>2. Memasukkan <i>keyword</i> = lam</li> <li>3. Memasukkan kategori pencarian = Diagnosa Keluar</li> <li>4. Menekan tombol Cari</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang mempunyai Diagnosa Keluar yang memiliki <i>value</i> lam
<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang mempunyai Diagnosa Keluar yang memiliki <i>value</i> lam
<b>Status</b>	Valid

### 6.2.22 Pengujian Validasi Tambah Data Pengkajian Asuhan Kebidanan

#### 1. Kasus Uji Tambah Data Pengkajian Asuhan Kebidanan

**Tabel 6.36 Kasus Uji Tambah Data Pengkajian Asuhan Kebidanan**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Pengkajian Asuhan Kebidanan
<b>Kode Kebutuhan</b>	RM_F2200
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Pengkajian Asuhan Kebidanan</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Memasukkan ruang rawat = Melati I</li> <li>4. Memasukkan tanggal = 07/15/2018</li> <li>5. Memasukkan jam = 12.31 AM</li> <li>6. Memasukkan riwayat penyakit = Hipertensi, DM</li> <li>7. Memasukkan riwayat penyakit keluarga = Hipertensi, DM</li> <li>8. Memasukkan status sosial = cemas, takut</li> <li>9. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data pengkajian asuhan kebidanan dan menampilkan halaman kembali data pengkajian asuhan kebidanan
<b>Hasil</b>	Sistem berhasil menyimpan data pengkajian asuhan kebidanan dan menampilkan halaman kembali data pengkajian asuhan kebidanan
<b>Status</b>	Valid

## 2. Kasus Uji Tambah Data Pengkajian Asuhan Kebidanan Alternatif 1

**Tabel 6.37 Kasus Uji Tambah Data Pengkajian Asuhan Kebidanan Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Pengkajian Asuhan Kebidanan Alternatif 1
<b>Kode Kebutuhan</b>	RM_F2200
<b>Prosedur</b>	1. Mengakses halaman Pengkajian Asuhan Kebidanan 2. Menekan tombol Tambah Data 3. Mengosongkan ruang rawat 4. Mengosongkan tanggal 5. Memasukkan jam = 12.31 AM 6. Memasukkan riwayat penyakit = Hipertensi, DM 7. Memasukkan riwayat penyakit keluarga = Hipertensi, DM 8. Memasukkan status sosial = cemas, takut 9. Menekan tombol Simpan
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

**6.2.23 Pengujian Validasi Lihat Data Pengkajian Asuhan Kebidanan**

## 1. Kasus Uji Lihat Data Pengkajian Asuhan Kebidanan

**Tabel 6.38 Kasus Uji Lihat Data Pengkajian Asuhan Kebidanan**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Pengkajian Asuhan Kebidanan
<b>Kode Kebutuhan</b>	RM_F2300
<b>Prosedur</b>	1. <i>Login sebagai Perekam Medis</i> 2. Mengakses halaman AKBID
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data pengkajian asuhan kebidanan
<b>Hasil</b>	Sistem menampilkan halaman data pengkajian asuhan kebidanan
<b>Status</b>	Valid

**6.2.24 Pengujian Validasi Lihat Detail Data Pengkajian Asuhan Kebidanan**

## 1. Kasus Uji Lihat Detail Data Pengkajian Asuhan Kebidanan

**Tabel 6.39 Kasus Uji Lihat Detail Data Pengkajian Asuhan Kebidanan**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data Pengkajian Asuhan Kebidanan
-----------------------	---

<b>Kode Kebutuhan</b>	RM_F2400
<b>Prosedur</b>	1. <i>Login</i> sebagai <i>Perekam Medis</i> 2. Mengakses halaman AKBID 3. Menekan tombol Buka
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data pengkajian asuhan kebidanan
<b>Hasil</b>	Sistem menampilkan detail data pengkajian asuhan kebidanan
<b>Status</b>	Valid

### 6.2.25 Pengujian Validasi Edit Data Pengkajian Asuhan Kebidanan

#### 1. Kasus Uji Edit Data Pengkajian Asuhan Kebidanan

**Tabel 6.40 Kasus Uji Edit Data Pengkajian Asuhan Kebidanan**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Pengkajian Asuhan Kebidanan
<b>Kode Kebutuhan</b>	RM_F2500
<b>Prosedur</b>	1. Mengakses halaman Pengkajian Asuhan Kebidanan 2. Menekan tombol edit 3. Melakukan <i>update</i> keadaan umum = suhu tubuh normal, tidak ada tanda-tanda vital yang mengganggu 4. Menekan tombol <i>Update</i>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data pengkajian asuhan kebidanan yang telah di <i>update</i> dan menampilkan kembali halaman data pengkajian asuhan kebidanan
<b>Hasil</b>	Sistem berhasil menyimpan data pengkajian asuhan kebidanan yang telah di <i>update</i> dan menampilkan kembali halaman data pengkajian asuhan kebidanan
<b>Status</b>	Valid

### 6.2.26 Pengujian Validasi Cari Data Pengkajian Asuhan Kebidanan

#### 1. Kasus Uji Cari Data Pengkajian Asuhan Kebidanan

**Tabel 6.41 Kasus Uji Cari Data Pengkajian Asuhan Kebidanan**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Pengkajian Asuhan Kebidanan
<b>Kode Kebutuhan</b>	RM_F2600
<b>Prosedur</b>	1. Mengakses halaman Pengkajian Asuhan Kebidanan 2. Memasukkan <i>keyword</i> = mua

	3. Memasukkan kategori pencarian = Keluhan Utama 4. Menekan tombol Cari
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang mempunyai Keluhan Utama yang memiliki <i>value</i> mua
<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang mempunyai Keluhan Utama yang memiliki <i>value</i> mua
<b>Status</b>	Valid

### 6.2.27 Pengujian Validasi Tambah Data Pengkajian Asuhan Keperawatan

#### 1. Kasus Uji Tambah Data Pengkajian Asuhan Keperawatan

**Tabel 6.42 Kasus Uji Tambah Data Pengkajian Asuhan Keperawatan**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Pengkajian Asuhan Keperawatan
<b>Kode Kebutuhan</b>	RM_F2700
<b>Prosedur</b>	1. Mengakses halaman Pengkajian Asuhan Keperawatan 2. Menekan tombol Tambah Data 3. Memasukkan tanggal pengkajian = 07/15/2018 4. Memasukkan jam pengkajian = 03.15 PM 5. Memasukkan keluhan saat ini = tidak ada 6. Memasukkan respirasi = TAK 7. Memasukkan riwayat penyakit = Asthma, Pneumonia 8. Menekan tombol Simpan
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data pengkajian asuhan keperawatan dan menampilkan halaman kembali data pengkajian asuhan keperawatan
<b>Hasil</b>	Sistem berhasil menyimpan data pengkajian asuhan keperawatan dan menampilkan halaman kembali data pengkajian asuhan keperawatan
<b>Status</b>	Valid

#### 2. Kasus Uji Tambah Data Pengkajian Asuhan Keperawatan Alternatif 1

**Tabel 6.43 Kasus Uji Tambah Data Pengkajian Asuhan Keperawatan Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Pengkajian Asuhan Keperawatan Alternatif 1
-----------------------	--

<b>Kode Kebutuhan</b>	RM_F2700
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Pengkajian Asuhan Keperawatan</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Mengosongkan tanggal pengkajian</li> <li>4. Mengosongkan jam pengkajian</li> <li>5. Memasukkan keluhan saat ini = tidak ada</li> <li>6. Memasukkan respirasi = TAK</li> <li>7. Memasukkan riwayat penyakit = Asthma, Pneumonia</li> <li>8. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

#### 6.2.28 Pengujian Validasi Lihat Data Pengkajian Asuhan Keperawatan

1. Kasus Uji Lihat Data Pengkajian Asuhan Keperawatan

**Tabel 6.44 Kasus Uji Lihat Data Pengkajian Asuhan Keperawatan**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Pengkajian Asuhan Keperawatan
<b>Kode Kebutuhan</b>	RM_F2800
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. <i>Login</i> sebagai <i>Perekam Medis</i></li> <li>2. Mengakses halaman AKPER</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data pengkajian asuhan keperawatan
<b>Hasil</b>	Sistem menampilkan halaman data pengkajian asuhan keperawatan
<b>Status</b>	Valid

#### 6.2.29 Pengujian Validasi Lihat Detail Data Pengkajian Asuhan Keperawatan

1. Kasus Uji Lihat Detail Data Pengkajian Asuhan Keperawatan

**Tabel 6.45 Kasus Uji Lihat Detail Data Pengkajian Asuhan Keperawatan**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data Pengkajian Asuhan Keperawatan
<b>Kode Kebutuhan</b>	RM_F2900
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. <i>Login</i> sebagai <i>Perekam Medis</i></li> <li>2. Mengakses halaman AKPER</li> <li>3. Menekan tombol Buka</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data pengkajian asuhan keperawatan

<b>Hasil</b>	Sistem menampilkan detail data pengkajian asuhan keperawatan
<b>Status</b>	Valid

### 6.2.30 Pengujian Validasi Edit Data Pengkajian Asuhan Keperawatan

#### 1. Kasus Uji Edit Data Pengkajian Asuhan Keperawatan

**Tabel 6.46 Kasus Uji Edit Data Pengkajian Asuhan Keperawatan**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Pengkajian Asuhan Keperawatan
<b>Kode Kebutuhan</b>	RM_F3000
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Pengkajian Asuhan Keperawatan</li> <li>2. Menekan tombol edit</li> <li>3. Melakukan <i>update</i> kesadaran = setengah sadar</li> <li>4. Melakukan <i>update</i> pasien mengerti tentang penyakitnya = tidak</li> <li>5. Menekan tombol <i>Update</i></li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data pengkajian asuhan keperawatan yang telah di <i>update</i> dan menampilkan kembali halaman data pengkajian asuhan keperawatan
<b>Hasil</b>	Sistem berhasil menyimpan data pengkajian asuhan keperawatan yang telah di <i>update</i> dan menampilkan kembali halaman data pengkajian asuhan keperawatan
<b>Status</b>	Valid

### 6.2.31 Pengujian Validasi Cari Data Pengkajian Asuhan Keperawatan

#### 1. Kasus Uji Cari Data Pengkajian Asuhan Keperawatan

**Tabel 6.47 Kasus Uji Cari Data Pengkajian Asuhan Keperawatan**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Pengkajian Asuhan Keperawatan
<b>Kode Kebutuhan</b>	RM_F3100
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Pengkajian Asuhan Keperawatan</li> <li>2. Memasukkan <i>keyword</i> = sad</li> <li>3. Memasukkan kategori pencarian = Kesadaran</li> <li>4. Menekan tombol Cari</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang mempunyai Kesadaran yang memiliki <i>value</i> sad



<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang mempunyai Kesadaran yang memiliki <i>value</i> sad
<b>Status</b>	Valid

### 6.2.32 Pengujian Validasi Tambah Data MR3.1

#### 1. Kasus Uji Tambah Data MR3.1

**Tabel 6.48 Kasus Uji Tambah Data MR3.1**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data MR3.1
<b>Kode Kebutuhan</b>	RM_F3200
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman MR3.1</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Memasukkan tanggal = 07/15/2018</li> <li>4. Memasukkan No RM Bayi = NN2131</li> <li>5. Memasukkan jk = p</li> <li>6. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data MR3.1 dan menampilkan halaman kembali data MR3.1
<b>Hasil</b>	Sistem berhasil menyimpan data MR3.1 dan menampilkan halaman kembali data MR3.1
<b>Status</b>	Valid

#### 2. Kasus Uji Tambah Data MR3.1 Alternatif 1

**Tabel 6.49 Kasus Uji Tambah Data MR3.1 Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data MR3.1 Alternatif 1
<b>Kode Kebutuhan</b>	RM_F3200
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman MR3.1</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Memasukkan tanggal = 07/15/2018</li> <li>4. Mengosongkan No RM Bayi</li> <li>5. Memasukkan jk = p</li> <li>6. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

### 6.2.33 Pengujian Validasi Lihat Data MR3.1

1. Kasus Uji Lihat Data MR3.1

**Tabel 6.50 Kasus Uji Lihat Data MR3.1**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data MR3.1
<b>Kode Kebutuhan</b>	RM_F3300
<b>Prosedur</b>	1. <i>Login sebagai Perekam Medis</i> 2. Mengakses halaman MR3.1
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data MR3.1
<b>Hasil</b>	Sistem menampilkan halaman data MR3.1
<b>Status</b>	Valid

### 6.2.34 Pengujian Validasi Lihat Detail Data MR3.1

1. Kasus Uji Lihat Detail Data MR3.1

**Tabel 6.51 Kasus Uji Lihat Data Detail MR3.1**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data MR3.1
<b>Kode Kebutuhan</b>	RM_F3400
<b>Prosedur</b>	1. <i>Login sebagai Perekam Medis</i> 2. Mengakses halaman MR3.1 3. Menekan tombol Buka
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data MR3.1
<b>Hasil</b>	Sistem menampilkan detail data MR3.1
<b>Status</b>	Valid

### 6.2.35 Pengujian Validasi Edit Data MR3.1

1. Kasus Uji Edit Data MR3.1

**Tabel 6.52 Kasus Uji Edit Data MR3.1**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data MR3.1
<b>Kode Kebutuhan</b>	RM_F3500
<b>Prosedur</b>	1. Mengakses halaman MR3.1 2. Menekan tombol edit 3. Melakukan <i>update</i> riwayat penyakit = tidak ada 4. Menekan tombol <i>Update</i>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data MR3.1 yang telah di <i>update</i> dan menampilkan kembali halaman data MR3.1
<b>Hasil</b>	Sistem berhasil menyimpan data MR3.1 yang telah di <i>update</i> dan menampilkan kembali halaman data MR3.1

Status	Valid
--------	-------

### 6.2.36 Pengujian Validasi Cari Data MR3.1

1. Kasus Uji Cari Data MR3.1

**Tabel 6.53 Kasus Uji Cari Data MR3.1**

Nama Kasus Uji	Kasus Uji Cari Data MR3.1
Kode Kebutuhan	RM_F3600
Prosedur	<ol style="list-style-type: none"> <li>1. Mengakses halaman MR3.1</li> <li>2. Memasukkan <i>keyword</i> = gus</li> <li>3. Memasukkan kategori pencarian = Refleks Fisiologi</li> <li>4. Menekan tombol Cari</li> </ol>
Hasil yang diharapkan	Sistem akan menampilkan data pasien yang mempunyai Refleks Fisiologi yang memiliki <i>value</i> gus
Hasil	Sistem berhasil menampilkan data pasien yang mempunyai Refleks Fisiologi yang memiliki <i>value</i> gus
Status	Valid

### 6.2.37 Pengujian Validasi Tambah Data MR3.2

1. Kasus Uji Tambah Data MR3.2

**Tabel 6.54 Kasus Uji Tambah Data MR3.2**

Nama Kasus Uji	Kasus Uji Tambah Data MR3.2
Kode Kebutuhan	RM_F3700
Prosedur	<ol style="list-style-type: none"> <li>1. Mengakses halaman MR3.2</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Memasukkan no reg = 00001</li> <li>4. Memasukkan tgl lahir bayi= 08/13/2018</li> <li>5. Memasukkan pukul = 09.20 AM</li> <li>6. Memasukkan jenis kelamin = p</li> <li>7. Memasukkan jenis kelahiran = tunggal</li> <li>8. Menekan tombol Simpan</li> </ol>
Hasil yang diharapkan	Sistem akan menyimpan data MR3.2 dan menampilkan halaman kembali data MR3.2
Hasil	Sistem berhasil menyimpan data MR3.2 dan menampilkan halaman kembali data MR3.2
Status	Valid

## 2. Kasus Uji Tambah Data MR3.2 Alternatif 1

Tabel 6.55 Kasus Uji Tambah Data MR3.2 Alternatif 1

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data MR3.2 Alternatif 1
<b>Kode Kebutuhan</b>	RM_F3700
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman MR3.2</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Mengosongkan no reg</li> <li>4. Memasukkan tgl lahir bayi= 08/13/2018</li> <li>5. Memasukkan pukul = 09.20 AM</li> <li>6. Memasukkan jenis kelamin = p</li> <li>7. Mengosongkan jenis kelahiran</li> <li>8. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

## 6.2.38 Pengujian Validasi Lihat Data MR3.2

## 1. Kasus Uji Lihat Data MR3.2

Tabel 6.56 Kasus Uji Lihat Data MR3.2

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data MR3.2
<b>Kode Kebutuhan</b>	RM_F3800
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Login sebagai <i>Perekam Medis</i></li> <li>2. Mengakses halaman MR3.2</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data MR3.2
<b>Hasil</b>	Sistem menampilkan halaman data MR3.2
<b>Status</b>	Valid

## 6.2.39 Pengujian Validasi Lihat Detail Data MR3.2

## 1. Kasus Uji Lihat Detail Data MR3.2

Tabel 6.57 Kasus Uji Lihat Detail Data MR3.2

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data MR3.2
<b>Kode Kebutuhan</b>	RM_F3900
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Login sebagai <i>Perekam Medis</i></li> <li>2. Mengakses halaman MR3.2</li> <li>3. Menekan tombol Buka</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data MR3.2
<b>Hasil</b>	Sistem menampilkan detail data MR3.2
<b>Status</b>	Valid

#### 6.2.40 Pengujian Validasi Edit Data MR3.2

1. Kasus Uji Edit Data MR3.2

**Tabel 6.58 Kasus Uji Edit Data MR3.2**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data MR3.2
<b>Kode Kebutuhan</b>	RM_F4000
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman MR3.2</li> <li>2. Menekan tombol edit</li> <li>3. Melakukan <i>update</i> lain-lain = keadaan bayi normal, proses persalinan berjalan lancar</li> <li>4. Menekan tombol <i>Update</i></li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data MR3.2 yang telah di <i>update</i> dan menampilkan kembali halaman data MR3.2
<b>Hasil</b>	Sistem berhasil menyimpan data MR3.2 yang telah di <i>update</i> dan menampilkan kembali halaman data MR3.2
<b>Status</b>	Valid

#### 6.2.41 Pengujian Validasi Cari Data MR3.2

1. Kasus Uji Cari Data MR3.2

**Tabel 6.59 Kasus Uji Cari Data MR3.2**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data MR3.2
<b>Kode Kebutuhan</b>	RM_F4100
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman MR3.2</li> <li>2. Memasukkan <i>keyword</i> = mat</li> <li>3. Memasukkan kategori pencarian = Kelainan</li> <li>4. Menekan tombol Cari</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang mempunyai Kelainan yang memiliki <i>value</i> mat
<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang mempunyai Kelainan yang memiliki <i>value</i> mat
<b>Status</b>	Valid

#### 6.2.42 Pengujian Validasi Tambah Data Rawat Jalan

1. Kasus Uji Tambah Data Rawat Jalan

**Tabel 6.60 Kasus Uji Tambah Data Rawat Jalan**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Rawat Jalan
<b>Kode Kebutuhan</b>	RM_F4200
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Rawat Jalan</li> <li>2. Menekan tombol Tambah Data</li> </ol>

	3. Memasukkan tanggal periksa = 17/15/2018 4. Memasukkan bb = 66 kg 5. Memasukkan tinggi badan = 165 kg 6. Memasukkan tekanan darah = 90/60 7. Memasukkan lingkaran lengan atas = 10 cm 8. Memasukkan nama dokter = dr. Retno B, Sp. OG 9. Memasukkan keterangan = control 2 minggu kemudian 10. Menekan tombol Simpan
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data rawat jalan dan menampilkan halaman kembali data rawat jalan
<b>Hasil</b>	Sistem berhasil menyimpan data rawat jalan dan menampilkan halaman kembali data rawat jalan
<b>Status</b>	Valid

## 2. Kasus Uji Tambah Data Rawat Jalan Alternatif 1

**Tabel 6.61 Kasus Uji Tambah Data Rawat Jalan Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Rawat Jalan Alternatif 1
<b>Kode Kebutuhan</b>	RM_F4200
<b>Prosedur</b>	1. Mengakses halaman Rawat Jalan 2. Menekan tombol Tambah Data 3. Memasukkan tanggal periksa = 17/15/2018 4. Mengosongkan bb 5. Memasukkan tinggi badan = 165 kg 6. Memasukkan tekanan darah = 90/60 7. Mengosongkan lingkaran lengan atas 8. Memasukkan nama dokter = dr. Retno B, Sp. OG 9. Mengosongkan keterangan 10. Menekan tombol Simpan
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

### 6.2.43 Pengujian Validasi Lihat Data Rawat Jalan

#### 1. Kasus Uji Lihat Data Rawat Jalan



Tabel 6.62 Kasus Uji Lihat Data Rawat Jalan

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Rawat Jalan
<b>Kode Kebutuhan</b>	RM_F4300
<b>Prosedur</b>	1. <i>Login</i> sebagai <i>Perekam Medis</i> 2. Mengakses halaman Rawat Jalan
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data rawat jalan
<b>Hasil</b>	Sistem menampilkan halaman data rawat jalan
<b>Status</b>	Valid

#### 6.2.44 Pengujian Validasi Lihat Detail Data Rawat Jalan

1. Kasus Uji Lihat Detail Data Rawat Jalan

Tabel 6.63 Kasus Uji Lihat Detail Data Rawat Jalan

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data Rawat Jalan
<b>Kode Kebutuhan</b>	RM_F4400
<b>Prosedur</b>	1. <i>Login</i> sebagai <i>Perekam Medis</i> 2. Mengakses halaman Rawat Jalan 3. Menekan tombol Buka
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data rawat jalan
<b>Hasil</b>	Sistem menampilkan detail data rawat jalan
<b>Status</b>	Valid

#### 6.2.45 Pengujian Validasi Edit Data Rawat Jalan

1. Kasus Uji Edit Data Rawat Jalan

Tabel 6.64 Kasus Uji Edit Data Rawat Jalan

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Rawat Jalan
<b>Kode Kebutuhan</b>	RM_F4500
<b>Prosedur</b>	1. Mengakses halaman Rawat Jalan 2. Menekan tombol edit 3. Melakukan <i>update</i> hasil USG = upload gambar 4. Menekan tombol <i>Update</i>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data rawat jalan yang telah di <i>update</i> dan menampilkan kembali halaman data rawat jalan
<b>Hasil</b>	Sistem berhasil menyimpan data rawat jalan yang telah di <i>update</i> dan menampilkan kembali halaman data rawat jalan
<b>Status</b>	Valid

### 6.2.46 Pengujian Validasi Cari Data Rawat Jalan

#### 1. Kasus Uji Cari Data Rawat Jalan

**Tabel 6.65 Kasus Uji Cari Data Rawat Jalan**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Rawat Jalan
<b>Kode Kebutuhan</b>	RM_F4600
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Rawat Jalan</li> <li>2. Memasukkan <i>keyword</i> = mam</li> <li>3. Memasukkan kategori pencarian = Nama Dokter</li> <li>4. Menekan tombol Cari</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang mempunyai Nama Dokter yang memiliki <i>value</i> mam
<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang mempunyai Nama Dokter yang memiliki <i>value</i> mam
<b>Status</b>	Valid

### 6.2.47 Pengujian Validasi Tambah Data Sebab Kematian

#### 1. Kasus Uji Tambah Data Sebab Kematian

**Tabel 6.66 Kasus Uji Tambah Data Sebab Kematian**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Sebab Kematian
<b>Kode Kebutuhan</b>	RM_F4700
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Sebab Kematian</li> <li>2. Memasukkan tanggal kematian = 17/15/2019</li> <li>3. Memasukkan penyakit a = komplikasi jantung</li> <li>4. Memasukkan penyakit lain = TB dan gula</li> <li>5. Memasukkan nama yang memberi keterangan sebab kematian (Nama Dokter) = dr. Retno B, Sp. OG</li> <li>6. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data sebab kematian dan menampilkan kembali data sebab kematian
<b>Hasil</b>	Sistem berhasil menyimpan data sebab kematian dan menampilkan kembali data sebab kematian
<b>Status</b>	Valid

#### 2. Kasus Uji Tambah Data Sebab Kematian Alternatif 1

Tabel 6.67 Kasus Uji Tambah Data Sebab Kematian Alternatif 1

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Sebab Kematian Alternatif 1
<b>Kode Kebutuhan</b>	RM_F4700
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Sebab Kematian</li> <li>2. Memasukkan tanggal kematian = 17/15/2019</li> <li>3. Memasukkan penyakit a = komplikasi jantung</li> <li>4. Mengosongkan penyakit lain</li> <li>5. Memasukkan nama yang memberi keterangan sebab kematian (Nama Dokter) = dr. Retno B, Sp. OG</li> <li>6. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

#### 6.2.48 Pengujian Validasi Lihat Data Sebab Kematian

1. Kasus Uji Lihat Data Sebab Kematian

Tabel 6.68 Kasus Uji Lihat Data Sebab Kematian

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Sebab Kematian
<b>Kode Kebutuhan</b>	RM_F4800
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. <i>Login</i> sebagai <i>Perekam Medis</i></li> <li>2. Mengakses halaman Sebab Kematian</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data sebab kematian
<b>Hasil</b>	Sistem menampilkan halaman data sebab kematian
<b>Status</b>	Valid

#### 6.2.49 Pengujian Validasi Cetak Laporan Persalinan

1. Kasus Uji Cetak Laporan Persalinan

Tabel 6.69 Kasus Uji Cetak Laporan Persalinan

<b>Nama Kasus Uji</b>	Kasus Uji Cetak Laporan Persalinan
<b>Kode Kebutuhan</b>	RM_F4900
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. <i>Login</i> sebagai <i>Perekam Medis</i></li> <li>2. Mengakses halaman Laporan Persalinan</li> <li>3. Memasukkan unit pelayanan = RSIAPF</li> <li>4. Memasukkan tgl1 = 08/01/2018</li> </ol>

	5. Memasukkan tgl2 = 09/01/2018 6. Memasukkan nama pelapor = Nuranisa Qalbi 7. Menekan tombol Cetak
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan <i>preview</i> halaman cetak laporan persalinan dalam bentuk pdf
<b>Hasil</b>	Sistem berhasil menampilkan <i>preview</i> halaman cetak laporan persalinan dalam bentuk pdf
<b>Status</b>	Valid

## 2. Kasus Uji Cetak Laporan Persalinan Alternatif 1

**Tabel 6.70 Kasus Uji Cetak Laporan Persalinan Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Cetak Laporan Persalinan Alternatif 1
<b>Kode Kebutuhan</b>	RM_F4900
<b>Prosedur</b>	1. <i>Login</i> sebagai Perekam Medis 2. Mengakses halaman Laporan Persalinan 3. Mengosongkan unit pelayanan 4. Memasukkan tgl1 = 08/01/2018 5. Memasukkan tgl2 = 09/01/2018 6. Memasukkan nama pelapor = Nuranisa Qalbi 7. Menekan tombol Cetak
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

## 6.2.50 Pengujian Validasi Cetak Laporan Kematian

### 1. Kasus Uji Cetak Laporan Kematian

**Tabel 6.71 Kasus Uji Cetak Laporan Kematian**

<b>Nama Kasus Uji</b>	Kasus Uji Cetak Laporan Kematian
<b>Kode Kebutuhan</b>	RM_F5000
<b>Prosedur</b>	1. <i>Login</i> sebagai Perekam Medis 2. Mengakses halaman Laporan Kematian 3. Memasukkan unit pelayanan = RSIAPF 4. Memasukkan tgl1 = 08/01/2018 5. Memasukkan tgl2 = 09/01/2018 6. Memasukkan nama pelapor = Nuranisa Qalbi 7. Menekan tombol Cetak

<b>Hasil yang diharapkan</b>	Sistem akan menampilkan <i>preview</i> halaman cetak laporan kematian dalam bentuk pdf
<b>Hasil</b>	Sistem berhasil menampilkan <i>preview</i> halaman cetak laporan kematian dalam bentuk pdf
<b>Status</b>	Valid

## 2. Kasus Uji Cetak Laporan Kematian Alternatif 1

**Tabel 6.72 Kasus Uji Cetak Laporan Kematian Alternatif 1**

<b>Nama Kasus Uji</b>	Kasus Uji Cetak Laporan Kematian Alternatif 1
<b>Kode Kebutuhan</b>	RM_F5000
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. <i>Login</i> sebagai Perekam Medis</li> <li>2. Mengakses halaman Laporan Kematian</li> <li>3. Mengosongkan unit pelayanan</li> <li>4. Memasukkan tgl1 = 08/01/2018</li> <li>5. Memasukkan tgl2 = 09/01/2018</li> <li>6. Memasukkan nama pelapor = Nuranisa Qalbi</li> <li>7. Menekan tombol Cetak</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Hasil</b>	Sistem berhasil menampilkan peringatan untuk mengisi <i>field</i> yang <i>required</i>
<b>Status</b>	Valid

## 6.2.51 Pengujian Validasi Tambah Data Visite

### 1. Kasus Uji Tambah Data Visite

**Tabel 6.73 Kasus Uji Tambah Data Visite**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Visite
<b>Kode Kebutuhan</b>	RM_F5100
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Visite Dokter</li> <li>2. Menekan tombol Tambah Data</li> <li>3. Memasukkan nama dokter = dr. Retno B, Sp. OG</li> <li>4. Memasukkan tanggal pemeriksaan = 08/08/2018</li> <li>5. Memasukkan tekanan darah = 120/70</li> <li>6. Memasukkan tindakan = Non Tindakan</li> <li>7. Menekan tombol Simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data visite dan menampilkan halaman kembali data visite

<b>Hasil</b>	Sistem berhasil menyimpan data visite dan menampilkan halaman kembali data visite
<b>Status</b>	Valid

### 6.2.52 Pengujian Validasi Lihat Data Visite

1. Kasus Uji Lihat Data Visite

**Tabel 6.74 Kasus Uji Lihat Data Visite**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Data Visite
<b>Kode Kebutuhan</b>	RM_F5200
<b>Prosedur</b>	1. <i>Login</i> sebagai Perawat 2. Mengakses halaman Visite Dokter
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman data visite
<b>Hasil</b>	Sistem menampilkan halaman data visite
<b>Status</b>	Valid

### 6.2.53 Pengujian Validasi Lihat Detail Data Visite

1. Kasus Uji Lihat Detail Data Visite

**Tabel 6.75 Kasus Uji Lihat Detail Data Visite**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Detail Data Visite
<b>Kode Kebutuhan</b>	RM_F5300
<b>Prosedur</b>	1. <i>Login</i> sebagai Perawat 2. Mengakses halaman Visite Dokter 3. Menekan tombol Buka
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan detail data visite
<b>Hasil</b>	Sistem menampilkan detail data visite
<b>Status</b>	Valid

### 6.2.54 Pengujian Validasi Edit Data Visite

1. Kasus Uji Edit Data Visite

**Tabel 6.76 Kasus Uji Edit Data Visite**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Visite
<b>Kode Kebutuhan</b>	RM_F5400
<b>Prosedur</b>	1. Mengakses halaman Visite 2. Menekan tombol edit 3. Melakukan <i>update</i> tindakan = Non Tindakan Medis 4. Menekan tombol <i>Update</i>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data visite yang telah di <i>update</i> dan menampilkan kembali halaman data visite



<b>Hasil</b>	Sistem berhasil menyimpan data visite yang telah di <i>update</i> dan menampilkan kembali halaman data visite
<b>Status</b>	Valid

### 6.2.55 Pengujian Validasi Cari Data Visite

#### 1. Kasus Uji Cari Data Visite

**Tabel 6.77 Kasus Uji Cari Data Visite**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Visite
<b>Kode Kebutuhan</b>	RM_F5500
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman Visite</li> <li>2. Memasukkan <i>keyword</i> = tin</li> <li>3. Memasukkan kategori pencarian = Tindakan</li> <li>4. Menekan tombol Cari</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data visite yang mempunyai Tindakan yang memiliki <i>value</i> tin
<b>Hasil</b>	Sistem berhasil menampilkan data visite yang mempunyai Tindakan yang memiliki <i>value</i> tin
<b>Status</b>	Valid

### 6.2.56 Pengujian Validasi Tambah Data Dokter

#### 1. Kasus Uji Tambah Data Dokter

**Tabel 6.78 Kasus Uji Tambah Data Dokter**

<b>Nama Kasus Uji</b>	Kasus Uji Tambah Data Dokter
<b>Kode Kebutuhan</b>	RM_F5600
<b>Prosedur</b>	<ol style="list-style-type: none"> <li>1. Mengakses halaman data dokter</li> <li>2. Memasukkan nama dokter = dr. Anshari Syakir, Sp.Ak</li> <li>3. Memasukkan alamat = Pampang</li> <li>4. Memasukkan no. telp = 08114417522</li> <li>5. Memasukkan jadwal praktik = Senin – Kamis 19.00</li> <li>6. Menekan tombol simpan</li> </ol>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data dokter
<b>Hasil</b>	Sistem berhasil menyimpan data dokter
<b>Status</b>	Valid

### 6.2.57 Pengujian Validasi Lihat Daftar Data Dokter

#### 1. Kasus Uji Lihat Daftar Data Dokter

**Tabel 6.79 Kasus Uji Lihat Daftar Data Dokter**

<b>Nama Kasus Uji</b>	Kasus Uji Lihat Daftar Data Dokter
<b>Kode Kebutuhan</b>	RM_F5700
<b>Prosedur</b>	1. <i>Login</i> sebagai Perekam Medis 2. Mengakses halaman daftar data dokter
<b>Hasil yang diharapkan</b>	Sistem akan menampilkan halaman daftar data dokter
<b>Hasil</b>	Sistem menampilkan halaman daftar data dokter
<b>Status</b>	Valid

### 6.2.58 Pengujian Validasi Edit Data Dokter

#### 1. Kasus Uji Edit Data Dokter

**Tabel 6.80 Kasus Uji Edit Data Dokter**

<b>Nama Kasus Uji</b>	Kasus Uji Edit Data Dokter
<b>Kode Kebutuhan</b>	RM_F5800
<b>Prosedur</b>	1. Mengakses halaman data dokter 2. Menekan tombol edit 3. Melakukan <i>update</i> alamat = Jl. Muna no. 35 Makassar 4. Melakukan <i>update</i> jadwal praktik = Senin – Kamis 19.00 – 22.00 Jumat – Minggu Libur 5. Menekan tombol <i>Update</i>
<b>Hasil yang diharapkan</b>	Sistem akan menyimpan data dokter yang telah di <i>update</i> dan menampilkan kembali halaman daftar data dokter
<b>Hasil</b>	Sistem berhasil menyimpan data dokter yang telah di <i>update</i> dan menampilkan kembali halaman daftar data dokter
<b>Status</b>	Valid

### 6.2.59 Pengujian Validasi Cari Data Dokter

#### 1. Kasus Uji Cari Data Dokter

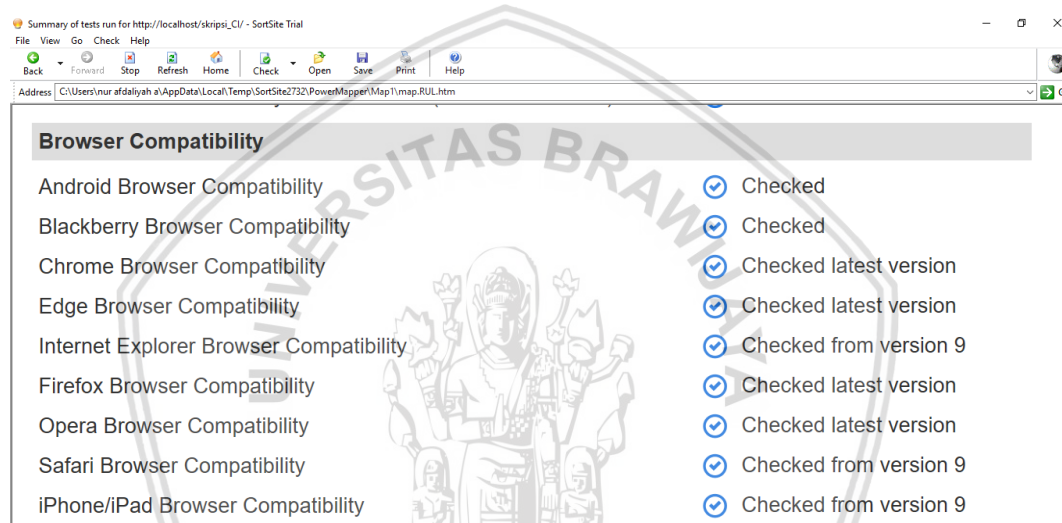
**Tabel 6.81 Kasus Uji Cari Data Dokter**

<b>Nama Kasus Uji</b>	Kasus Uji Cari Data Dokter
<b>Kode Kebutuhan</b>	RM_F5900
<b>Prosedur</b>	1. Mengakses halaman data dokter 2. Memasukkan <i>keyword</i> = pam

<b>Hasil yang diharapkan</b>	Sistem akan menampilkan data pasien yang memiliki nama dr. Anshari Syakir, Sp. OG
<b>Hasil</b>	Sistem berhasil menampilkan data pasien yang memiliki nama dr. Anshari Syakir, Sp. OG
<b>Status</b>	Valid

### 6.3 Pengujian *Compatibility*

Pengujian *compatibility* dilaksanakan untuk mengetahui kemampuan sistem aplikasi rekam medis RSIAPF dapat dijalankan pada beberapa *browser*, *versi*, dan *platform*. Pengujian *compatibility* dilakukan menggunakan piranti lunak SortSite.



**Gambar 6.4 Hasil Pengujian *Compatibility***

Gambar 6.4 adalah hasil pengujian *compatibility* dengan menggunakan piranti lunak SortSite. Hasil dari pengujian *compatibility* yang dilakukan terhadap sistem aplikasi rekam medis menunjukkan bahwa sistem dapat berjalan pada 9 versi *browser* yang sudah disebutkan pada Gambar 6.4

## BAB 7 PENUTUP

### 7.1 Kesimpulan

Dari hasil penelitian pembangunan Aplikasi Rekam Medis RSIAPF berbasis web, dapat diperoleh kesimpulan sebagai berikut:

1. Berdasarkan hasil analisis kebutuhan sistem diperoleh 59 kebutuhan fungsional dan 1 kebutuhan non-fungsional. Pada tahap analisis kebutuhan sistem, dilakukan pemodelan analisis data yang berbentuk *Entity Relationship Diagram* (ERD) serta pemodelan kebutuhan dalam bentuk *use case diagram* dan *use case scenario*.
2. Berdasarkan hasil perancangan, diperoleh perancangan arsitektur, perancangan data, perancangan komponen, dan perancangan antarmuka. Perancangan arsitektur dibentuk menjadi *sequence diagram* dan *class diagram* dimana *sequence diagram* menjelaskan urutan proses serta interaksi antar objek dalam fungsional sistem, sedangkan *class diagram* menggambarkan perancangan kelas yang menggunakan pola perancangan MVC (*Model-View-Controller*). Perancangan data dibuat dalam bentuk *Physical Data Model* (PDM) sesuai dengan table yang ada pada *database*. Perancangan komponen dibuat dalam bentuk *pseudocode* yang akan digunakan pada implementasi. Dan perancangan antarmuka menghasilkan 15 tampilan yang menggambarkan antarmuka setiap halaman yang ada pada aplikasi rekam medis.
3. Berdasarkan hasil implementasi menggunakan *Object-Oriented-Programming* (OOP), diperoleh spesifikasi pengembangan sistem yang menjelaskan spesifikasi perangkat keras dan perangkat lunak yang digunakan dalam melakukan implementasi sistem, implementasi data dalam bentuk *Data Definition Language* (DDL), implementasi kode program yang diperoleh dari hasil perancangan komponen, serta implementasi antarmuka yang diperoleh dari perancangan antarmuka.
4. Berdasarkan hasil pengujian, pengujian unit dilakukan dengan metode *white-box testing* yang menghasilkan nilai 100% valid untuk semua jalur uji. Pengujian validasi yang dilakukan dengan metode *black-box testing* terhadap 50 kebutuhan fungsional yang menghasilkan nilai 100% valid untuk semua kasus uji. Serta pengujian *compatibility* dengan menggunakan piranti lunak *SortSite* diperoleh informasi bahwa sistem dapat berjalan dengan baik pada 100% dari 9 versi *browser* yang diujikan.

### 7.2 Saran

Saran yang dapat diberikan untuk pengembangan aplikasi rekam medis selanjutnya adalah:

1. Sistem dapat mengolah data administrasi pasien dan hasil laboratorium.
2. Sistem dapat membuat surat rujukan secara komputerisasi

## DAFTAR PUSTAKA

- Aljufri, A., 2013. Aplikasi Rekam Medis Studi Kasus Klinik Universitas Widyatama. *Universitas Widyatama Bandung*.
- Anon., 2008. *Keputusan Menteri Kesehatan (KEPMENKES) RI No 129/MENKES/SK/II/2008 tentang Rekam Medis*. Jakarta: Kementerian Kesehatan.
- Anon., 2008. *Peraturan Menteri Kesehatan (PERMENKES) No. 269/MENKES/PER/III/2008 tentang Rekam Medis*. Jakarta: Kementerian Kesehatan.
- Arief, B., 2016. Aplikasi Rekam Medis Pasien Berbasis Web pada Klinik Belinda Rajeg kab. Tangerang.
- Dawood, R., 2017. Aplikasi Rekam Medis Studi Kasus Klinik Universitas Widyatama. *Universitas Syiah Kuala*.
- Dika, Y., 2014. *Rekomendasi Izin Operasional RSIA*. Prof. M. Farid. Makassar: s.n.
- EMS, T., 2016. *All in One Web Programming: Pemrograman Web dengan HTML, PHP, AJAX, dan jQuery Mobile*. Jakarta: Elex Media Komputindo.
- Hatta, G. R., 2012. *Pedoman Manajemen Informasi Kesehatan di Sarana Pelayanan Kesehatan*. Jakarta: UI Press.
- Junaedi, F., 2005. *Panduan Lengkap Pemrograman PHP untuk Membuat WEB Dinamis*. Yogyakarta: PD. Anindya.
- McLeod, R., 2008. *Sistem Informasi Manajemen*. 10 ed. Jakarta: Selemba Empat.
- Medik, D. J. B. P., 2006. *Pedoman Penyelenggaraan Prosedur Rekam Medis Rumah Sakit di Indonesia Revisi II*. Jakarta: Departemen Kesehatan RI.
- Organization, W. H., 2001. *Medical Records Manual: A Guide For Developing Countries*. Manila: WHO Library Cataloguing in Publication Data.
- Puspitasari, 2011. *Pemrograman Web Database dengan PHP dan MySql*. Jakarta: Skripta.
- Raharjo, B., 2011. *Membuat Database Menggunakan MySql*. Bandung: Informatika.
- Raharjo, B., 2018. *Belajar Otodidak Framework Codeigniter*. Revisi ed. Bandung: Informatika.
- S, B. S. & Sakinah, F., 2008. *Rekam Medis Terkomputerisasi*. Jakarta: UI-Press.
- Shalahuddin, M. & S, R. A., 2013. *Rekayasa Perangkat Lunak: Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Simamarta, J., 2010. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi Offset.

- S, L., 2012. Unified Model Language (UML) for Database System and Computer Applications. *Int J. Database Theory Appl*, 05(01), pp. 157-164.
- Sommerville, I., 2011. *Software Engineering*. 9th ed. United States of America: Addison-Wesley.
- Sulhan, M., 2007. *Pengembangan Aplikasi Berbasis Web dengan PHP & ASP*. Yogyakarta: Gava Media.
- Wimmie, H., 2009. Rekam Medis Elektronik. Volume 02, pp. 36-41.

